

AD-A091 686

CARNEGIE-MELLON UNIV PITTSBURGH PA DEPT OF PSYCHOLOGY
ACQUISITION OF PROBLEM-SOLVING SKILL.(U)

F/G 5/10

OCT 80 J R ANDERSON, J G GREENO, P J KLINE

N00014-78-C-0725

UNCLASSIFIED

TR-80-5

NL

1 (P) 1
AU
AC79-186

END
DATE
FILMED
12 80
DTIC

LEVEL

12

AD A091686

Acquisition of Problem-Solving Skill

DTIC
SELECTED
NOV 7 1980
C

John R. Anderson
Carnegie-Mellon University

James G. Greeno
University of Pittsburgh

Paul J. Kline and David M. Neves
Carnegie-Mellon University

Approved for public release; distribution unlimited.
Reproduction in whole or in part is permitted for any purpose
of the United States government.

This research was supported by the Personnel and Training Research Programs, Psychological Services Division, Office of Naval Research, under Contract No.: N00014-78-C-0725, Contract Authority Identification Number, NR No.: 154-399 to John Anderson and under Contract No.: N00014-78-C-0022, Contract Authority Identification Number, NR No.: 157-408 to James Greeno.

DDC FILE COPY

80 11 05 010

unclassified

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER Technical Report #80-5	2. GOVT ACCESSION NO. AD-A091 686	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) Acquisition of Problem-Solving Skill	5. TYPE OF REPORT & PERIOD COVERED Final technical	
6. PERFORMING ORG. REPORT NUMBER		7. AUTHOR(s) John R./Anderson, James G./Greeno, Paul J./Kline, David M./Neves
8. CONTRACT OR GRANT NUMBER(s) N00014-78-C-0725		9. PERFORMING ORGANIZATION NAME AND ADDRESS Department of Psychology Carnegie-Mellon University Pittsburgh, Pennsylvania 15213
10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS RR042-04-01 NR 154-399 66153N		11. CONTROLLING OFFICE NAME AND ADDRESS Personnel and Training Research Programs Office of Naval Research Arlington, Virginia 22217
12. REPORT DATE 7 October 1980		13. NUMBER OF PAGES 72
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)		15. SECURITY CLASS. (of this report) unclassified
15a. DECLASSIFICATION/DOWNGRADING SCHEDULE		16. DISTRIBUTION STATEMENT (of this Report) Approved for public release; distribution unlimited.
17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)		
18. SUPPLEMENTARY NOTES To appear in J.R. Anderson (Ed.), <u>Cognitive Skills and their Acquisition</u> , Hillsdale, N.J.: Lawrence Erlbaum Associates, 1981.		
19. KEY WORDS (Continue on reverse side if necessary and identify by block number)		
Geometry	Schemata	Representation
Mathematics education	Planning	Composition
Skill acquisition	Proofs	Proceduralization
Learning	Problem-solving	Analogy
Production systems	Search	Generalization
Discrimination	Automatization	Declarative encodings
Procedural encodings	Practice effects	
20. ABSTRACT (Continue on reverse side if necessary and identify by block number) The paper is organized into five major sections and a summary. Each of the five major sections is devoted to a different aspect of our analysis of proof skills in geometry and their acquisition. The first section presents an analysis of how a student searches for a proof in geometry after he has acquired his basic skills. This provides a basic framework within which to understand each of the types of learning that we then discuss in the remaining four sections. The second section is concerned with what we call text learning. It is concerned with		

unclassified

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

unclassified

SECURITY CLASSIFICATION OF THIS PAGE(When Data Entered)

20. Abstract (continued)

what the student directly encodes from the text, and more importantly, the processes that use this direct encoding to guide the problem solving. The third section is concerned with a process that we call subsumption, a means by which the student encodes new information into existing knowledge structures. Fourth, we discuss the processes of knowledge compilation by which knowledge is transformed from its initial encoded form, which is declarative, to a more effective procedural form. The final major section discusses how practice serves to tune and optimize the skill so that the proof search is performed more successfully.

19. Key Words (continued)

Tuning
Partial matching

Accession For	
NTIS GRA&I	<input checked="checked" type="checkbox"/>
DTIC TAB	<input type="checkbox"/>
Unannounced	
Justification	
By _____	
Distribution/	
Availability Codes	
Dist	Special
A	

unclassified

SECURITY CLASSIFICATION OF THIS PAGE(When Data Entered)

Introduction

This chapter presents some recent explorations we have made in the domain of learning to solve geometry proof problems. The background of these explorations consisted of two research programs: one investigating general principles of learning and the other investigating the nature of problem-solving skill in geometry. We have been conducting these two programs relatively independently since about 1976, but both programs have used versions of Anderson's (1976) ACT production system as a computational formalism. Our investigations of learning have included a discussion of general assumptions about learning and design issues (Anderson, Kline & Beasley, 1980) and an analysis of prototype formation (Anderson, Kline & Beasley, 1979). Our studies of geometry problem solving have included an analysis of forms of knowledge required for successful performance (Greeno, 1978), a discussion of goal representation (Greeno, 1976), and a discussion of schematic knowledge involved in planning (Greeno, Magone & Chaiklin, 1979). The knowledge required to solve proof problems is moderately complex, but since its structure has been studied in some detail it serves as a feasible target for a theoretical analysis of learning. In our investigations, we have found ways to extend both the previous analysis of learning and the previous analysis of problem-solving skill significantly.

The paper is organized into five major sections and a summary. Each of the five major sections is devoted to a different aspect of our analysis of proof skills in geometry and their acquisition. The first section will present an analysis of how a student searches for a proof in geometry after he has acquired his basic skills. This will provide a basic framework within which to understand each of the types of learning that we will then discuss in the remaining four sections. The second section is concerned with what we call text learning. It is concerned with what the student directly encodes from the text, and more importantly, the processes that use this direct encoding to guide the problem solving. The third section is concerned with a process that we call subsumption, a means by which the student encodes new information into existing knowledge structures. Fourth, we discuss the processes of knowledge compilation by which knowledge is transformed from its initial encoded form, which is declarative, to a more effective procedural form. The final major section discusses how practice serves to tune and optimize the skill so that the proof search is performed more successfully.

The empirical information used in this research came from a number of sources. First, a 14-year-old student agreed to study geometry with our supervision and observation. This student, to whom we refer as Subject R, met with one or two of us on quite a regular basis for 30 sessions covering 2 months. In each session, lasting for about 45 minutes, R read sections of a geometry textbook and worked problems from the text, thinking aloud as he proceeded. This work done by R was his only study of geometry; his mathematics work in school was on algebra, and this represented accelerated instruction for him. We also had access to a set of three interviews obtained from a 14-year-old student who was beginning to work on geometry as his regular school mathematics study. The other major source of information was a set of interviews obtained from six students who were taking a regular course in geometry. These interviews were obtained on an approximately weekly basis throughout the course.

A Model of the Skill Underlying Proof Generation

Most successful attempts at proof generation can be divided into two major episodes -- an episode in which a student attempts to find a plan for the proof and an episode in which the student translates that plan into an actual proof. The first stage we call *planning* and the second *execution*. It is true that actual proof generation behavior often involves alternation back and forth between the two modes -- with the student doing a little planning, writing some proof, running into trouble, planning some more, writing some more proof, and so on. Still we believe that planning exists as a logically and empirically separable stage of proof generation. Moreover, we believe that planning is the more significant aspect and the aspect which is more demanding of learning. Execution, while not necessarily trivial, is more "mechanical".

It is also the case that planning tends to pass without comment from the student. (We had one subject who preferred to pass through this stage banging his hands on his forehead.) However, we have tried to open this stage up to analysis through the gathering of verbal protocols. These protocols indicate a lot of systematic goings-on which seem to fit under the title of planning.

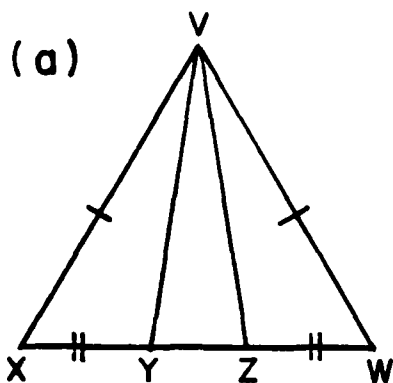
A plan, in the sense we are using it here, is an outline for action -- the action in this case being proof execution. We believe that the plan students emerge with is a specification of a set of geometric rules

that allows one to get from the givens of the problem, through intermediate levels of statements, to the to-be-proven statement. We call such a plan a *proof tree*. This idea about the general nature of a plan is consistent with other discussions of planning such as Greeno, Magone and Chaiklin (1979), Newell and Simon (1972), and Sacerdoti (1977).

Figure 1 illustrates (a) an example geometry problem and (b) a proof tree. In the tree, the goal to prove two angles congruent leads to the subgoal of proving the triangles $\triangle XVZ$ and $\triangle WVY$ congruent. This goal is achieved by the side-angle-side (SAS) postulate. The first side $\overline{VX} \cong \overline{VW}$ is gotten directly from the givens. Since these sides form an isosceles triangle, they also imply $\angle VXZ \cong \angle VWY$, the second part of the SAS congruence pattern. The third part $\overline{XZ} \cong \overline{WY}$ can be gotten from the other given that $\overline{XY} \cong \overline{WZ}$. A proof can be obtained from Figure 1 by unpacking various links in the proof tree. Such a proof is given below. It should be noted that some of these links map into multiple lines of proof. For instance, the link connecting $\overline{XY} \cong \overline{WZ}$ to $\overline{XZ} \cong \overline{WY}$ maps into the 9 lines 4-12 in the proof. This is one of the important reasons why we characterize the proof tree as an *abstract* specification of a proof.

1. $\overline{VX} \cong \overline{VW}$	given
2. $\overline{SY} \cong \overline{WZ}$	given
3. $\triangle XYZ$ is isosceles	definition
4. $\angle VXZ \cong \angle VWY$	base \angle 's of isosceles
5. $XY = WZ$	def. of \cong
6. $YZ = YZ$	symmetric property of equality
7. $\overline{XY} + \overline{YZ} = \overline{YZ} + \overline{WZ}$	addition property of equality
8. $XYZW$	given
9. $XZ = XY + YZ$	segment addition
10. $WY = YZ + WZ$	segment addition
11. $\overline{XZ} \cong \overline{WY}$	substitution
12. $\overline{XZ} \cong \overline{WY}$	def. of \cong
13. $\triangle XVZ \cong \triangle WVY$	SAS
14. $\angle XVZ \cong \angle WVY$	corresponding parts of congruent \triangle 's

The proof tree is, of course, not something that students typically draw out for themselves. Rather it is a knowledge structure in the head. Various remarks of students suggest to us that it is a real knowledge structure, not just a product of our theoretical fantasies. For instance, one student described a proof as "an upside down pyramid". (For a student a proof tree would be upside-down since the actual proof ends in the to-be-proven statement. However, we display the tree right-side up (to-be-proven statement at the top) because that arrangement facilitates theoretical discussion.)



Given: $\overline{VX} \cong \overline{VW}$, $\overline{XY} \cong \overline{WZ}$, \overline{XYZW}
 Prove: $\angle XVZ \cong \angle WVY$

(b)

PLAN

$$\angle XVZ \cong \angle WVY$$



$$\triangle ZVX \cong \triangle WVY$$

$$\angle VXZ \cong \angle VYW$$

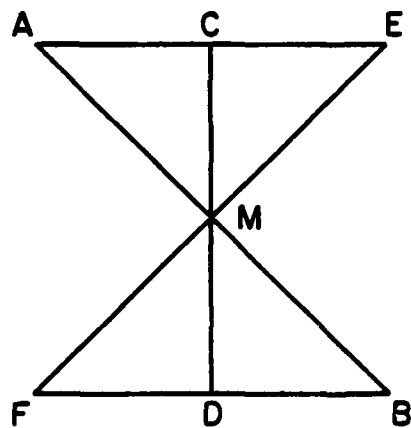
$$\overline{XZ} \cong \overline{WY}$$

$$\overline{VX} \cong \overline{VW}$$

$$\overline{XY} \cong \overline{WZ}$$

FIGURE 1

A problem with its proof tree and detailed proof.



Given: M is the midpoint of \overline{AB} and \overline{CD}
Prove: M is the midpoint of \overline{EF}

FIGURE 2

Problem for simulation of planning.

As a pedagogical aside, we might say that we think it very important that the central role of such abstract plans be taken into account in instruction. It is very easy to direct instruction to proof execution and ignore the underlying plan that guides the writing of the proof. The idea of focusing on the general structure of the proof, rather than on its details, agrees with Duncker's (1945) suggestion of attempting to present organic proofs, rather than mechanical ones. Since the proof tree is closely related to the main subgoals that arise during solution, instruction that used the proof tree explicitly might facilitate acquisition of useful strategic knowledge.

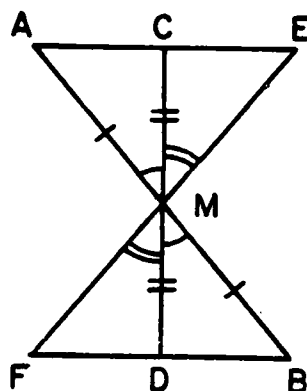
Creating a proof tree is not a trivial problem. The student must either try to search forward from the givens trying to find some set of paths that converge satisfactorily on the to-be-proven statement or he must try to search backward from the to-be-proven statement trying to find some set of dependencies that lead back to the givens. Using unguided forward or backward search, it is easy to become lost. We will argue that students use a mixture of forward and backward search. This mixture, along with various search heuristics they acquire, enables students to deal with the search demands of proof problems found in high school geometry texts.

Example: The Simulation

We will discuss an example problem taken from Chapter 4 of Jurgensen, Donnelly, Maier, and Rising (1975). This problem is shown in Figure 2. It is among the most difficult problems found in that chapter. We first discuss the way that an ACT simulation performed on this problem. This will serve to illustrate more fully our conception of the planning process in proof generation and how this planning is achieved in a production system. Then we will see how ACT's performance compares with that of a high school subject.

ACT's search for a proof tree involves simultaneously searching backward from the to-be-proven statement and searching forward from the givens. An attempt is made to try to bring these two searches together. This search process creates a network of logical dependencies. When successful ACT will eventually find in its search some set of logical dependencies that defines a satisfactory proof tree. This proof tree will be embedded within the search network.

Figure 3 illustrates the search at an early stage of its development. ACT has made forward



Goal M is midpoint
of \overline{EF}

$$\overline{EM} \cong \overline{FM}$$

$$\triangle CME \cong \triangle DMF$$

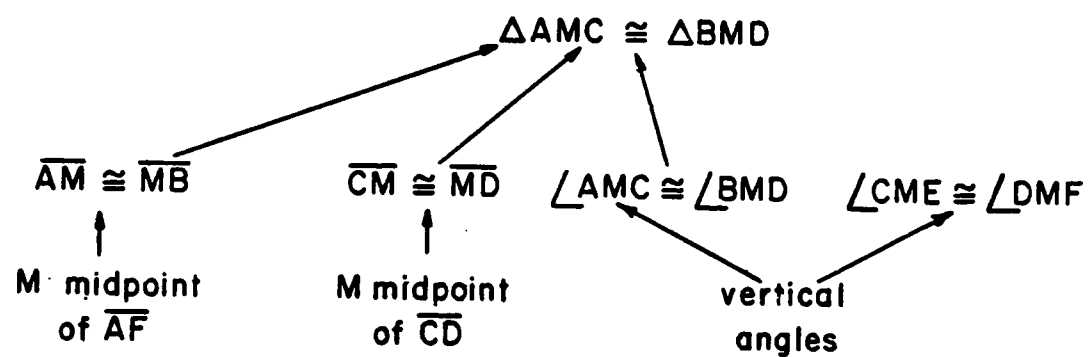


FIGURE 3

Problem net early in planning.

inferences that there are two sets of congruent angles $\angle AMC \cong \angle BMD$ and $\angle CME \cong \angle DMF$ because of vertical angles and that there are two sets of congruent segments, $\overline{AM} \cong \overline{MB}$ and $\overline{CM} \cong \overline{MD}$ because of the midpoint information. (These inferences were made by specific productions in the ACT simulation, but we are postponing discussion of productions until the learning sections.) With this information in hand ACT also makes the forward inference that $\triangle AMC \cong \triangle BMD$ because of the side-angle-side postulate. It has been our experience that almost everyone presented with this problem works forward to this particular inference as the first step to solving the problem. Note at this point that neither ACT nor our subjects know how this inference will fit into the final proof of the problem.

Meanwhile, ACT has begun to unwind a plan of backward inferences to achieve the goal. It has translated the midpoint goal into the goal of proving the congruence $\overline{EM} \cong \overline{FM}$. In turn it has decided to achieve this goal by proving that these two segments are corresponding parts of congruent triangles, $\triangle CME$ and $\triangle DMF$. This means that it must prove that these two triangles are congruent -- its new subgoal.

Note that the forward inferences have progressed much more rapidly than the backward inferences. This is because backward inferences, manipulating a single goal, are inherently serial whereas the forward inferences can apply in parallel. With respect to the serial-parallel issue it should be noted that the backward and forward searches progress in parallel.

Figure 3 illustrates the limit to the forward inferences that ACT generates. While there are, of course, more forward inferences that could be made, this is the limit to the inferences for which ACT has rules strong enough to apply.

Figure 4 illustrates the history of ACT's efforts to reason backward to establish that $\triangle CME \cong \triangle DMF$. ACT first attempts to achieve this by the side-side-side (SSS) postulate (a basically random choice at this stage of learning). This effort is doomed to failure because the triangle congruence has been set as a subgoal of proving one of the sides congruent. When ACT gets to the goal of establishing $\overline{EM} \cong \overline{FM}$ it recognizes the problem and backs away. Our subject R, like ACT, had a propensity to plunge into hopeless paths. One component of learning (discrimination) would eventually stop it from setting such hopeless subgoals.

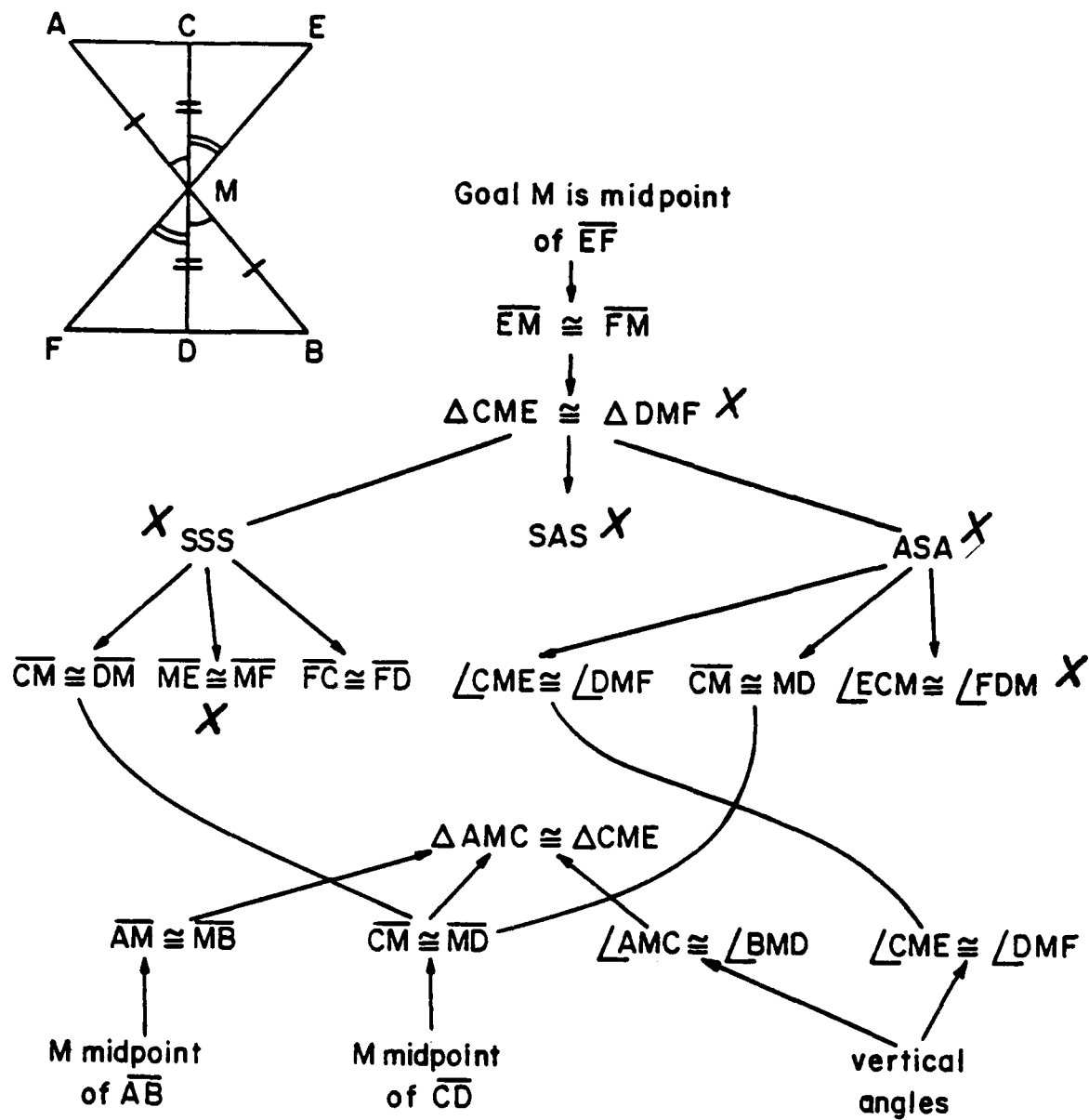


FIGURE 4

Trace of some backward-chaining efforts by ACT.

We will skip over ACT's unsuccessful attempt to achieve the triangle congruence by side-angle-side (SAS) and look in detail at its efforts with the angle-side-angle (ASA) postulate. Two of the three pieces required for this -- $\angle CME \cong \angle DMF$ and $\overline{CM} \cong \overline{MD}$ have already been established by forward inferences. This leaves the third piece to be established -- that $\angle ECM \cong \angle FDM$. This can be inferred by supplementary angles from something that is already known -- that $\triangle AMC \cong \triangle BMD$. However, ACT does not have the postulate for making this inference available. This corresponds to a blindness of our subject R with respect to using supplementary angles. Although the opportunity did not arise in this problem because he was following a different path to solution, many other times he overlooked opportunities to achieve his goals by the supplementary angle rule.

Having failed the three available methods for proving triangle congruence, ACT backed up and found a different pair of triangles, $\triangle AME$ and $\triangle BMF$, whose triangle congruence would establish the higher goal that $\overline{EM} \cong \overline{FM}$. (It turns out that, by failing on the supplementary angle needed to establish $\triangle CME \cong \triangle DMF$ and going on to try $\triangle AME \cong \triangle BMF$, ACT finds the shorter proof.)

Fortuitously, ACT chooses ASA as its first method. The attempt to apply this method is illustrated in Figure 5. A critical congruence, that $\angle AME \cong \angle BMF$ is gotten because these are vertical angles. Note that this inference was not made by the forward-reasoning vertical-angle rules. This is caused by a difficulty that the ACT pattern-matcher has in seeing that lines define multiple angles. The segments \overline{AM} and \overline{ME} that define $\angle AME$ were already used in extracting the angles $\angle AMC$ and $\angle CME$ for use by the forward reasoning vertical angle postulate.

ACT is also able to get the other parts of the ASA pattern. The side $\overline{AM} \cong \overline{BM}$ has already been gotten by forward inference. The fact that $\angle EAM \cong \angle FBM$ can be inferred from the fact that $\triangle AMC \cong \triangle BMD$ since the angles are corresponding parts of congruent triangles. With this ACT has found its proof tree embedded within the search net. That proof tree is highlighted in Figure 5.

Comparison of ACT to Subject R

It is of interest to see how ACT's behavior compares to that of a typical student. We have a more or less complete record of subject R's learning and work at geometry through Chapter 4 of Jurgensen, Donnelly, Maier, and Rising. In particular, we have a record of his performance on the critical

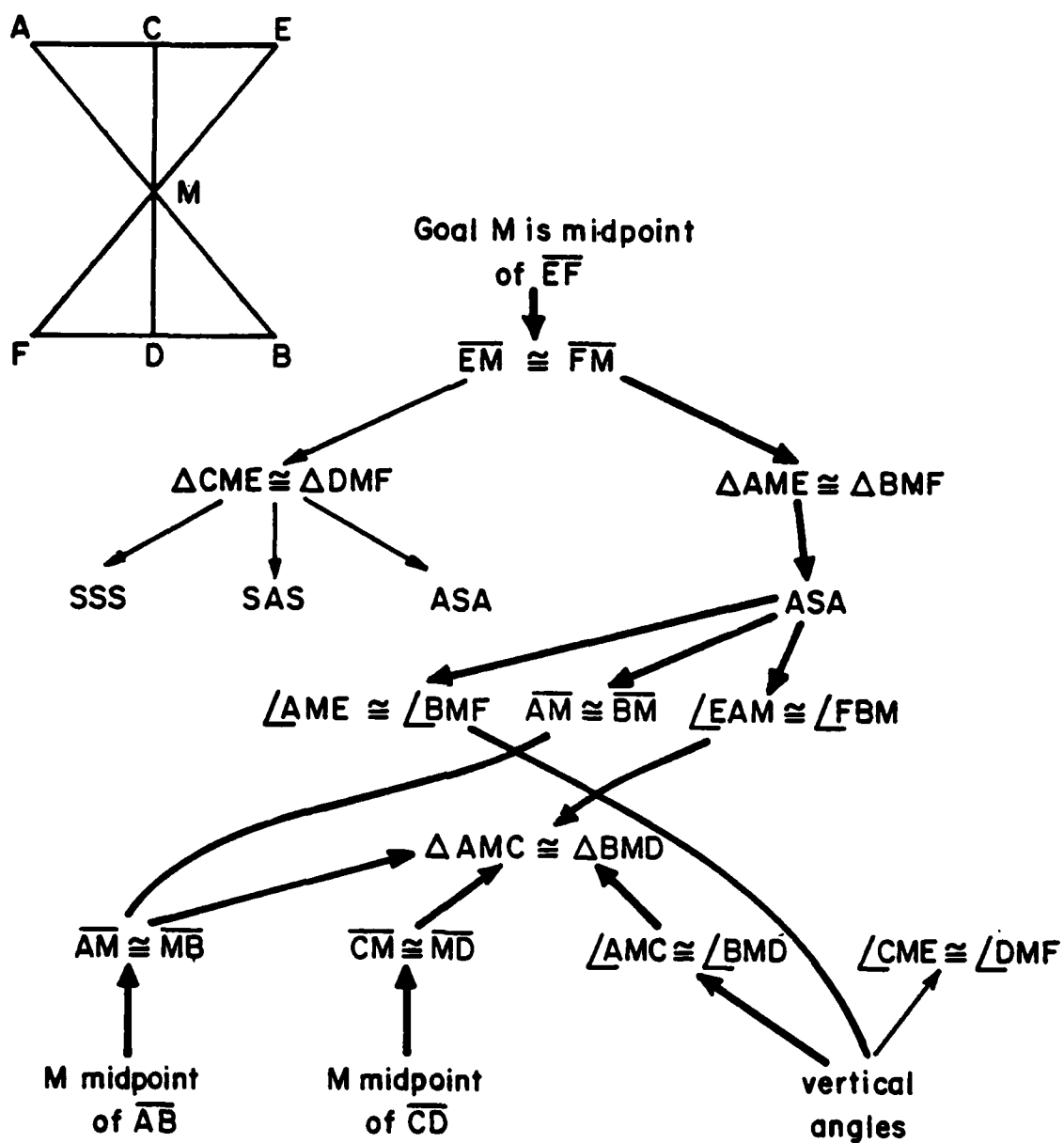


FIGURE 5

Application of ASA method by ACT.

problem in Figure 2.

Subject R's performance did not correspond to that of ACT in all details. This is to be expected because ACT's choices about what rules to apply have an important probabilistic component to them. However, we can still ask whether ACT and subject R made inferences of the same character. One way of answering this question is to determine whether ACT could have produced R's protocol if the probabilities came out correctly. By this criterion ACT is compatible with much of R's protocol.

Like ACT, R began by making the forward inferences necessary to conclude $\triangle AMC \cong \triangle BMD$ and then making this conclusion. Like ACT these inferences were made with little idea for how they would be used. Then like ACT, R began to reason backward from his goal to prove that M was the midpoint of \overline{EF} to the goal of proving triangle congruence. However, unlike ACT he was lucky and chose the triangle $\triangle AME \cong \triangle BMF$ first. Unlike ACT again, but this time unlucky, he first chose SAS as his method for establishing the triangle congruence. He got $\overline{AM} \cong \overline{MB}$ from previous forward inference and the $\angle EAM \cong \angle FBM$ from the fact that $\triangle AMC \cong \triangle BMD$ -- just as ACT obtained this in trying to use ASA. However, he then had to struggle with the goal of proving $\overline{AE} \cong \overline{BF}$. Unlike ACT, subject R was reluctant to back up and he tenaciously tried to find some way of achieving his goal. He was finally told by the instructor to try some other method. Then he turned to ASA. He already had two pieces of the rule by his efforts with SAS and quickly got the third component $\angle AME \cong \angle BMF$ from the fact that they were vertical angles. Note that subject R also failed to make this vertical angle inference in forward mode and only made it in backward mode -- as did ACT.

In conclusion, we think that R's behavior was very similar in character to that of ACT. The only major exception was R's reluctance to back up when a particular method was not working out.

When subject R worked on this problem he had spent a considerable number of hours studying geometry. Thus, the version of ACT that simulates his performance represents knowledge acquired over some history of learning, although we will discuss later some significant ways in which this version of ACT is less advanced than some others that have been programmed for geometry problem solving. We did not simulate the acquisition of skills represented in the version of ACT whose performance we have just described. However, we had certain learning processes in mind when we

developed the simulation program, and its design was intended to be a plausible hypothesis about skill that could be acquired with those learning processes, as well as to simulate performance like that of subject R. We have developed simulations of various components of the learning process that lead to a skill product of the kind we simulated in this version of ACT. It is to this learning analysis that we now turn.

Learning from the Text

Out of their initial interactions with the text and the teacher, students emerge with a rudimentary ability to solve proof problems in geometry. One might suppose that students are directly instructed as to what the procedures are that underlie proof generation. However, there is no such instruction in standard texts and yet students can learn from these texts. The texts provide only indirect information about how to generate proofs. We have been able to identify three sources of the knowledge that permits students their initial success in solving problems. These sources can be ordered on a continuum of sophistication and power. Lowest on this continuum, students use worked out examples as an analogical base for solving further problems.¹ Next on the continuum students can apply general problem solving methods to apply their understanding of the postulates and definitions of geometry. Highest on the continuum, students can extend general concepts that they have to incorporate the new material of geometry (subsumption). We will discuss the first two kinds of learning in this section and the other kind of learning in the next section. We organize the first two together because they involve more direct encoding of knowledge from the text. This issue here turns out to be, not encoding the knowledge, but rather how the knowledge is used once encoded. With respect to the third kind of learning, the subsumption processes underlying the encoding proves to be of considerable interest.

¹Another form of learning based on worked-out examples is possible, but the authors of this paper differ in our opinions about whether it is plausible. The possibility is that learners acquire new problem-solving procedures directly from observation of the solutions of example problems. We have worked out some simulations of learning based on this idea, but we do not describe them in this chapter, partly because of space limitations, and partly because we do not all agree that the mechanism of production designation used in that learning psychologically plausible.

Analogy: Use of Examples

A good deal of a typical student's time is spent studying worked out examples of proofs or going over examples that he/she has worked out. Students are able to commit a good bit of these examples to memory and what they cannot remember they refer back to the text for. Many students make heavy use of the examples (recalled from memory or read from the text) to guide solutions to new problems. Students are able to spontaneously detect similarities between past problems they remember and new problems they are facing. Once having detected the similarities, they enjoy fair success in mapping/transforming proofs for one problem to another problem. This use of prior examples we refer to as *analogy*, which seems an appropriate label and is consistent with others' uses of the term (Rumelhart & Norman, this volume; Winston, 1978). One such pair of problems for which analogy is successful is illustrated in Figure 6. The text provides a solution to Problem (a) and then presents Problem (b) as the first problem for proof in the section. Our subject R noticed the similarity between the two problems, went back to the first and almost copied over the solution.

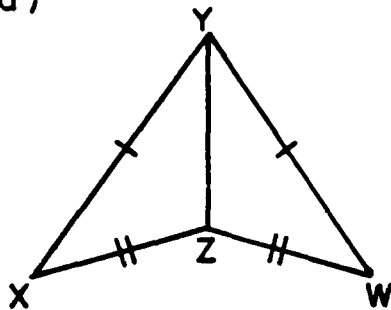
To account for successful solution of Problem (b) by analogy using Problem (a), we must assume that the student has a facility to partially match the diagram and given statements of one problem to the diagram and given statements of another problem. We have recently developed such partial matching facilities for the ACT theory.

One problem with analogy to specific problems is that it appears to be effective only in the short run because students' memory for specific problems tends to be short-lived. All examples we have of analogy in R's protocols come within the same section of a chapter. We have no examples of problems in one section reminding R of problems in an earlier section.

A second problem with pure analogy is that it is superficial. Any point of similarity between two problems increases the partial match. It is no accident that the two pairs of triangles in Figure 6 are oriented in the same direction, although this is completely irrelevant for the success of the analogy.

In ACT analogy depends on partial matching processes which are quite "syntactic" in character. That is, the partial matching processes just count up the degree of overlap in the problem descriptions without really evaluating whether the overlaps are essential to the appropriateness of the

(a)

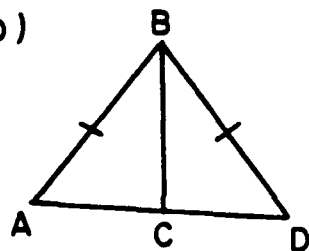


Given: $\overline{XY} \cong \overline{WY}$

$\overline{XZ} \cong \overline{WZ}$

Prove: $\triangle XYZ \cong \triangle WYZ$

(b)



Given: $\overline{AB} \cong \overline{DB}$

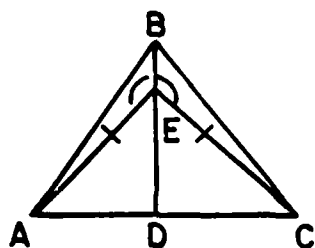
\overline{BC} bisects \overline{AD}

Prove: $\triangle ABC \cong \triangle DBC$

FIGURE 6

Two problems with obvious similarity.

(a)

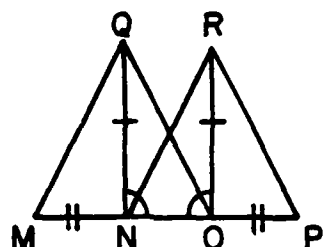


Given: $\overline{AE} \cong \overline{EC}$

$\angle BEA \cong \angle BEC$

Prove: $\triangle ABD \cong \triangle CBD$

(b)



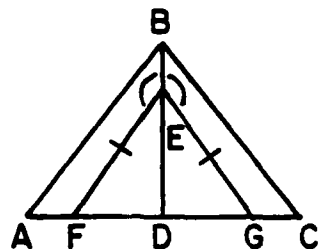
Given: $\overline{QN} \cong \overline{RN}$

$\angle QON \cong \angle RON$

$\overline{MN} \cong \overline{PN}$

Prove: $\triangle MQO \cong \triangle PRN$

(c)



Given: $\overline{AB} \cong \overline{BC}$

$\angle BEF \cong \angle BEG$

$\overline{AB} \parallel \overline{FE}$

$\overline{BC} \parallel \overline{EG}$

Prove: $\triangle ABD \cong \triangle CBD$

FIGURE 7

Problems illustrating limited validity of superficial analogy.

solution or not. In our own selves we note a tendency to respond to overlap between problems in this same superficial way. Consider the three problems in Figure 7. At a deep level the first two problems are really quite similar. Larger triangles contain smaller triangles. To prove the containing triangles congruent it is first necessary to prove the contained triangles congruent. The contained triangles in the two problems are congruent in basically the same way and they overlap with the containing triangles in basically the same way. However, on first glance the two problems seem quite different. In contrast, on first glance, the two problems in parts (a) and (c) of Figure 7 appear to have much in common. Now it is true that upon careful inspection we can determine that the first pair provides a more useful analogy than the second pair. However, it seems that the function of analogy in problem solving of this sort is that of a *noticing* function. Similar problems spontaneously come to mind as possible models for solutions. If the superficial similarity between Problems (a) and (b) is not sufficient for the analogy to be noticed there will never be the opportunity for careful inspection to realize how good the deep correspondence is.

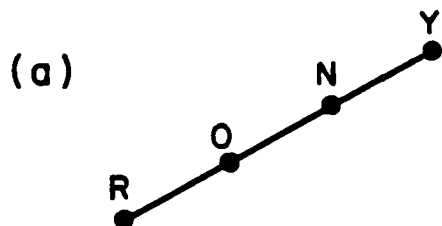
There is one very nice illustration of the problem with the superficiality of analogy in the protocol of R. This concerns a pair of problems that come in the first chapter. Figure 8 illustrates the two problems. Part (a) illustrates the initial problem R studied along with an outline of the proof. Later in the section R came across Problem (b) and immediately noticed the analogy. He tried to use the first proof as a model for how the second should be structured. Analogous to the first line $RO = NY$ he wrote down the line $AB > CD$. Then analogous to the second line $ON = ON$ he wrote down $BC > BCI$. His semantic sensitivities caught this before he went on and he abandoned the attempt to use the analogy.

Interpretation of Definitions and Postulates

We have found a schema-like representation to be very useful for representing a student's initial declarative encoding of some of the geometry postulates and definitions. (As we will explain shortly, we have also found schemata to be useful structures for encoding prior knowledge.) Table 1 illustrates a schema encoding for the SAS postulate which is stated in the text as:

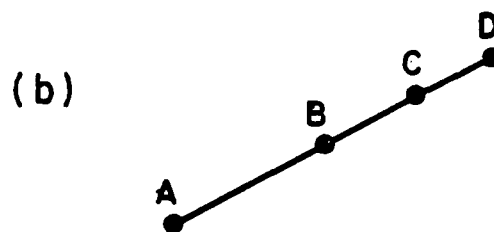
If two sides and the included angle of one triangle are congruent to the corresponding parts of another triangle, the triangles are congruent.

The diagram in Figure 9 accompanied this statement.



Given: $RO = NY$, \overline{RONY}
 Prove: $RN = OY$

$RO = NY$
 $ON = ON$
 $RO + ON = ON + NY$
 \overline{RONY}
 $RO + ON = RN$
 $ON + NY = OY$
 $RN = OY$



Given: $AB > CD$, \overline{ABCD}
 Prove: $AC > BD$

$AB = CD$
 $BC = BC$

!!!

FIGURE 8

A problem in which superficial analogy goes wrong.

Table 1

SAS Schema

Background

s1 is a side of $\triangle XYZ$
s2 is a side of $\triangle XYZ$
A1 is an angle of $\triangle XYZ$
?A1 is included by s1 and s2?
s3 is a side of $\triangle UVW$
s4 is a side of $\triangle UVW$
A2 is an angle of $\triangle UVW$
?A2 is included by s3 and s4?

Hypothesis

s1 is congruent to s3
s2 is congruent to s4
A1 is congruent to A2

Conclusion

$\triangle XYZ$ is congruent to $\triangle UVW$

Comment

This is the side-angle-side postulate

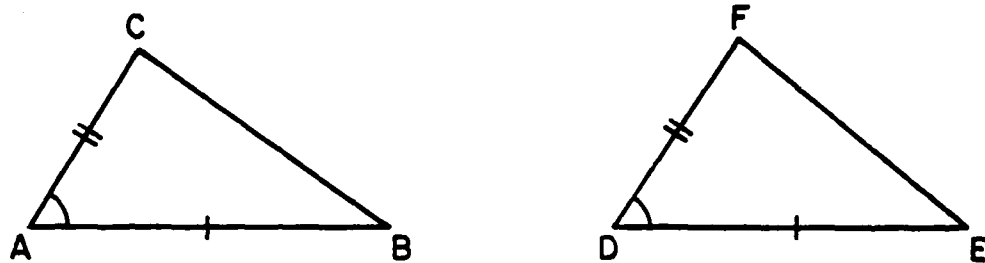


FIGURE 9

Diagram accompanying the SAS postulate.

We propose a student creates a declarative representation like the schema in Table 1. The schema in Table 1 is divided into four categories of information: background, hypothesis, conclusion, and comment. The background includes relevant descriptive information about the diagram and contains the constraints that allow the variables (sides and angles) to be properly bound. The hypothesis specifies propositions that must be asserted in previous statements of the proof. The comment contains additional information relevant to its use. Here we have the name of the postulate that prescribes what the student should write as a reason.

In reading such a postulate subject R would typically read through at a slow but constant rate and then go to the diagram trying to relate it to the statement of the postulate. More time would be spent looking at the diagram and relating it to the postulate statement than on anything else. We take this to indicate time spent extracting the background information which is not very saliently presented for a particular problem. Students are not always successful at extracting the relevant background information. For instance, subject R failed to appreciate what was meant by "included angles" (hence the question marks around these clauses in the clauses in the background in Table 1). It was only sometime later, after direct intervention by the experimenter, that R got this right.

We regard the knowledge structure in Table 1 to be schema-like in that it is a unit organized into parts according to "slots" like background, hypothesis, conclusion, and comment. The knowledge structure is declarative in that it can be used in multiple ways by interpretative procedures. For instance, the following production would be evoked to apply that knowledge in a working-backwards manner:

```

IF the goal is to prove a statement
   and there is a schema that has this statement as conclusion
THEN set as subgoals to match the background of the schema
   and after that to prove the hypothesis of the schema
  
```

If the schema is in working memory and its conclusion matches the current goal, this production will start up the application of the schema to the current problem. First the background is matched to bind the variables and then the hypotheses are checked.

A schema of this kind can be evoked in several ways in a working-forward mode. In this way, it is

similar to a constraint, in Steele and Sussman's (Note 5) sense. For instance, the following production would serve to evoke the schema in this manner:

```

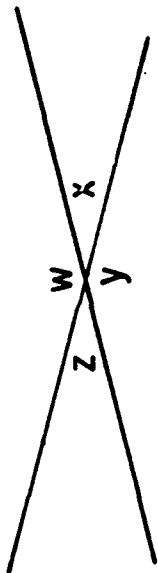
IF a particular statement is known to be true
  and there is a schema that includes this statement in its hypothesis
THEN set as subgoals to match the background of the schema
      and then to match the remaining hypotheses of the schema
      and if they match, to add the conclusion of the schema
  
```

This production is evoked when only part of the hypothesis is satisfied. Because there can be any number of statements in a hypothesis, it is not possible to have a general interpretive production that matches all the statements of any hypothesis (since a production only matches a fixed number of statements in its condition). Rather it is necessary to evoke on a partial match and then to check if the remaining statements match. This is one instance of many that illustrates the need for piecemeal application when knowledge is used interpretively. Before the rest of the hypotheses can be checked the background must be matched to bind variables. If the hypotheses do match, the conclusion is added as an inference.

Note that whether the knowledge is used in reasoning forward or reasoning backward, the background must be matched first. In reasoning forward, the hypotheses serve as a test of the applicability of the schema and the conclusion is added. In reasoning backwards, the conclusion serves as the test and the hypotheses are added as subgoals. However, in either mode the background serves as a precondition that must be satisfied.

Subsumption: Learning with Understanding

In this section we present an analysis of meaningful learning, in the sense that was used by Katona (1940) and Wertheimer (1945/1959). Discussions of meaningful learning have distinguished between learning that results in relatively mechanical skill and learning that results in understanding of the structure of the problem situation. We have developed a system that learns meaningfully by acquiring a structure in which problem-solving procedures are integrated with general concepts the system already has. The outcome of learning is a schema that provides a structure for understanding a problem situation in general terms, as well as guiding problem-solving performance. We believe that in working out this implementation we have reached a more specific and clearer understanding of the



$$w + x = 180^\circ$$

$$w + z = 180^\circ$$

or

$$w + z = w + x$$

$$x = z$$

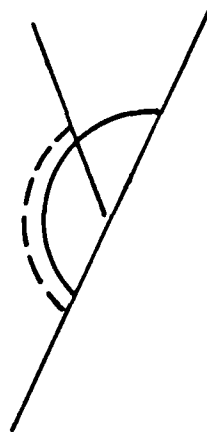
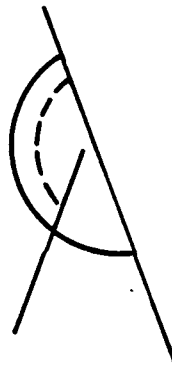
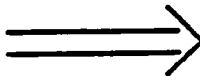
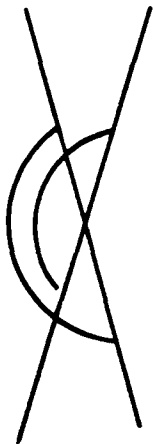


FIGURE 10

The Vertical Angles Problem, after Wertheimer (1945/1959).

processes and prior knowledge required for meaningful learning of problem-solving procedures.

Wertheimer and Katona gave numerous examples of the distinction between mechanical and meaningful learning. One of Wertheimer's is particularly relevant to our discussion. The problem is to prove that the opposite vertical angles in Figure 10 are equal: for example, to show $x = z$. There is a simple algebraic proof, shown in the lower left portion of Figure 10. Wertheimer argued that this algebraic solution could be learned without the student understanding the problem well. He advocated teaching in a way that would call attention to important spatial relationships in the problem, shown in the lower right portion of Figure 10. The idea that understanding is strengthened by appreciation of these spatial relationships seems intuitively compelling. There is a problem of specifying the nature of understanding in a way that clarifies how it is strengthened, and we hope that our discussion in this section contributes to an improvement in clarifying specificity of this issue. We will describe some general features of the system we developed to investigate meaningful learning and then describe the learning that this system accomplishes in two different learning tasks.

Schema-based problem solving

The system that we developed to investigate meaningful learning represents problems schematically. The schema used in debugging the system has the relationships between two objects that are joined together to form another object; we call the schema WHOLE/PARTS. The structure of a schema is consistent with recent proposals, especially of Bobrow and Winograd (1977) in their KRL system. A schema is used to represent relationships among a set of objects that fit into the schema's *slots*. The schema has some application conditions that are procedural in our system, implemented as *schematizing productions*. A schema also has *procedural attachments*, which are schematic descriptions of procedures that can be performed on objects of the kinds that the schema can be used to interpret. Finally, a schema has *contextual associations*, which provide information about features of the schema's application that vary in different contexts.

An example is in Table 2, where we show the main components of the schema in the state we used as initial knowledge prior to learning in the context of geometry problems. The schema provides a structure for understanding problems such as, "There are 5 girls and 3 boys; how many children are there?" or "There are 6 dogs and 8 animals; how many cats are there?" The schematizing production

Table 2

Slots:

PART 1, PART 2, WHOLE

Procedural Attachments:

COMBINE/CALCULATE
SEPARATE/CALCULATE
ADJUST/SAME-WHOLE

Contextual Associations:

SET → NUMBER

Schematizing Production

IF V1 is a set,
 and V2 is a set,
 and V3 is a set,
 and V1 is a kind of V3,
 and V2 is a kind of V3;
THEN Instantiate WHOLE/PARTS,
 with V1 in PART 1, V2 in PART 2,
 and V3 in WHOLE.

requires that two of the sets be related to the third through a "kind-of" relation, and uses these relations in determining which sets are the parts and which is the whole. (Other schematizing productions could easily be added; for example, sets that are located in separate places for problems such as "There are 3 cookies on the counter and 4 cookies in the jar; How many cookies are there in all?") More systematic studies of the use of schemata in understanding arithmetic word problems have been conducted by Heller (Note 2) and by Riley (Note 4).

One of the procedural attachments of the schema is shown in Table 3. This is organized in the way proposed by Sacerdoti (1977), with information that specifies prerequisites, consequences, and actions that are performed in order to execute the procedure. These procedural attachments are schema-like themselves and have some similarity to the types of schema discussed earlier (e.g. in Table 1).² In Table 3 Vrelation refers to a variable whose value is determined by the contextual association of the schema; for problems about sets, Vrelation would be the number associated with a set. Therefore, if the problem is about sets, the prerequisite of COMBINE/CALCULATE is that numbers of the parts are known, the consequence is that the number of the whole will be found, and the action to perform is addition of the known numbers.

The other procedural attachments in the WHOLE/PARTS schema are organized similarly. The procedure SEPARATE/CALCULATE has a prerequisite of known values associated with the whole and one of the parts, and the consequence of finding the other part. The procedure ADJUST/SAME-WHOLE is used when one of the parts is too large or too small for some reason, and a shift of members between the two parts is made in a way that keeps the value of the whole constant.

To solve problems using schemata, ACT has a strategy implemented as a set of productions similar to the productions on pp. 11. The specific strategy that we implemented for this system uses one step of backward chaining if it can; otherwise it is a forward-chaining strategy. ACT first tries to find a

²Our use of the term "schemata" is somewhat different here than in the earlier presentation (Table 1). This is similar to the general literature, where "schema" is used to refer to various kinds of structures; however, we may be unusual in adopting two different meanings in the same article. The difference reflects the fact that the implementations described in these two sections were in the hands of different authors (Table 1: Anderson; this section: Greeno). We believe that the two kinds of schematic structures are quite compatible. The background information in Table 1 correspond to the condition of a schematizing production here, the hypotheses in Table 1 to the prerequisites of procedural attachment, and the conclusion to the consequence of a procedural attachment.

Table 3

Prerequisites:

Vrelation of PART 1 is known.

Vrelation of PART2 is known.

Consequence:

Vrelation of WHOLE is found.

Performance:

Add numbers (Vrelation of PART 1, Vrelation of Part 2)

procedure whose consequences matches its current problem goal. If one is found, ACT checks the prerequisites of that procedure, and if they are satisfied ACT performs the actions of that procedure. If prerequisites are not satisfied, ACT looks for another procedure whose consequence matches the goal of the problem. Any time ACT fails to find a procedure with a consequence matching its current goal, it tests prerequisites of its procedures and attempts to work forward.

Learning tasks

Several different forms of learning can be considered in relation to schemata of the kind we are using. We will discuss three here. First, we simulated generalization of a schema to a new problem domain. The system initially had the schema shown in Table 2, which it could use to schematize problems about sets. It was shown an example problem involving segments, and learned a new schematizing production and contextual association enabling it to use the WHOLE/PARTS schema to schematize problems about segments.

A second task that we have simulated involved adding new structure to an existing schema. The added structures were new procedural attachments that enable ACT to solve a new kind of problem. Initially the only procedures attached to the WHOLE/PARTS schema involved numerical calculations or adjustments. ACT was shown an example problem involving proof, and acquired new procedures for writing steps in the proof. These new procedures were attached to ACT's WHOLE/PARTS schema, so that ACT's later solution of other proof problems included its general understanding of part-whole structural relations.

The third task for which we have performed a simulation involved synthesizing new schematic structure. A problem was presented for which the existing WHOLE/PARTS schema is too simple to provide a complete interpretation. The system acquired a more complex schema in which two WHOLE/PARTS structures are included as subschemata, and in which the previously learned proof procedures were available for use in solving more complex problems.

Task application in a new domain

Initially the WHOLE/PARTS schema was coded for the model and debugged on problems involving numbers of objects in different sets, such as the "There are 5 children and 2 boys; how many girls are

there?" A new example problem was presented to the model, consisting of a diagram consisting of a segment with endpoints A and C, and a point on the segment marked B. The following propositions were also presented: "Given: $AB = 5$, $BC = 3$; Find AC." The solution was also presented: " $AC = 5 + 3 = 8$." The model included a production for interpreting the given information as lengths of segments in the diagram. It also had a production for interpreting "Find AC" by setting a goal to find the length of that segment.

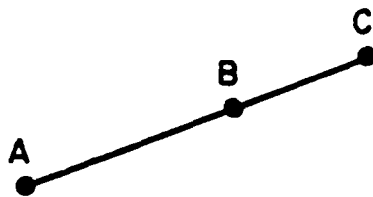
The new knowledge required for the schema to be applied in the domain of segments is a new schematizing production whose action assigns objects in the segment problem to slots in the WHOLE/PART schema. There is a difficulty with formulating such a production and that difficulty arises from the fact that the learner is not directly told to incorporate this situation as an instance of his WHOLE/PART schema. We assumed that activation of the appropriate schema would require some relatively specific cue, which is consistent with evidence that indicates that human problem solvers are unlikely to notice structural similarities among problems unless there are fairly obvious signals available (Gick & Holyoak, 1980). We provided the needed cue in the form of the WHOLE/PART schema's name; this might correspond to a situation in a class where the teacher points out that the whole segment is composed of other segments as its parts.

Our simulation identifies the schema that is named in the explanation of the solution-step that is presented. The respective slot-roles of the various objects in the problem are sorted out by examining the problem-solving action. A production is built whose performance matches the action that is shown, and the slots mentioned in the procedure are matched to the objects that are associated with the values in the action. The action of the new production is determined by these identifications: it refers to the objects in the problem that is to be schematized and associates them with the slots they should occupy.

Table 4 shows the schematizing production that the system learned. We also programmed the system to store a new contextual association, so that for future problems involving segments, the problem solver would use the lengths of segments as the relevant properties. Thus, by acquisition of the schematizing production the system is able to bring to bear its knowledge about WHOLE/PART

Table 4

IF	V112 is a segment and V116 is a segment and V120 is a segment and V120 contains V112 and V120 contains V116 and V119 and V121 are endpoints of V112 and V119 and V122 are endpoints of V116 and V121 and V122 are endpoints of V120
THEN	Instantiate WHOLE/PARTS, with V112 in PART1, and V116 in PART2, and V120 in WHOLE.



Given: \overline{ABC}

Prove: $AB = AC - BC$

Statement

1. \overline{ABC}
2. $AC = AB + BC$
3. $AB = AC - BC$

Reason

1. Given
2. Segment addition (Step 1)
3. Subtraction property (Step 2)

FIGURE 11

An example used to teach problem-solving procedures.

relationships to the solving of geometry problems.

We believe that the ability to solve PART/WHOLE problems of the kind we discussed here is probably learned well before students begin studying geometry, and that they probably understand the part-whole relationship involved. We have not observed any students in their work on geometry whose performance indicates a lack of the knowledge modeled in this section. On the other hand, the extension of this knowledge and understanding to the early problems of geometry is by no means automatic for all students, as we will discuss later.

Acquiring New Procedural Attachments

The second meaningful learning task that we simulated used the problem shown in Figure 11. The learning simulated in this situation was very similar to that discussed on pp. 11-13.

In this task, ACT began with the WHOLE/PARTS schema, and the schematizing production shown in Table 4, so it represented the part-whole structure of the segments in the diagram. Steps 2 and 3 represented new actions for ACT -- that is, did not have knowledge of procedures that would produce these proof steps. Using its representation of the problem situation and the statements in the proof steps, ACT stored new schemata descriptions of procedures and attached them to its WHOLE/PARTS schema. As a result, ACT had structures similar to the SAS schema shown in Table 1 corresponding to the segment-addition postulate and the subtraction property of equality. However, these new structures were integrated with ACT's general understanding of structures that involve part-whole relationships.

A consequence of attaching the segment-addition procedure to the WHOLE/PARTS schema is that it is available for use in more complicated problem situations, when the WHOLE/PARTS schema is invoked in ACT's understanding of more complex problems. We discuss an extension of that kind in the next section.

Building a new schema from old schemata

Another task that we have used to simulate learning is shown in Figure 12 (also discussed in Figure 8). Conceptual understanding of this problem involves seeing the given segment RY as two overlapping WHOLE/PARTS structures, each having a distinct part as well as a part that the two

Table 5

Slots

PARTA, PARTAB, PARTB, WHOLEA, WHOLEB, WHOLE

Subschemata

WHOLE/PARTSA, WHOLE/PARTSB

Procedural Attachments:

SUBSTITUTION

ADDITION-PROPERTY

SUBTRACTION-PROPERTY

Contextual Association:

Segment ----> Length

Schematizing Productions

IF V1, V2, V3, V4, V5, and V6 are segments,
and V6 includes V1, V2, V3, V4, and V5,
and V4 includes V1 and V2,
and V5 includes V2 and V3,

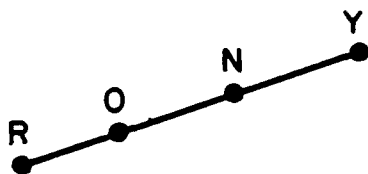
THEN Instantiate WHOLE/PARTS (= schema1) with V1 as PART1,
V2 as PART2, V4 as WHOLE;
and instantiate WHOLE/PARTS (= schema2) with V3 as PART1,
V2 as PART2, V5 as WHOLE;
and instantiate OVERLAP/WHOLE/PARTS with V1 as PARTA, V2 as PARTB
V3 as PARTB, V4 as WHOLEA, V5 as WHOLEB, V6 as WHOLE,
schema1 as WHOLE/PARTSA, and schema2 as WHOLE/PARTSB.

structures share. Table 5 shows a schema that we have partially implemented with the required structure. The important new idea in this simulation is the inclusion of subschemata as components of the schema. The main consequence is that the procedures already attached to the subschema are automatically available in situations schematized with the superschema. For example, the segment-addition procedure, attached to the WHOLE/PARTS schema because of the learning discussed just above, is available in solving problems that can be schematized with the OVERLAP/WHOLE/PARTS schema. In the problem-solving strategy that we implemented, the problem-solver first tries to apply procedures that are attached to the main schema, and when none are applicable, procedures attached to the subschemata are tested.

There are powerful consequences of schematizing line segment problems as instances of WHOLE/PARTS or OVERLAP/WHOLE/PARTS structures. At this level of representation a wide variety of problems have essentially the same character. For instance, consider the angle problem presented in Figure 13. From the point of view of the OVERLAP/WHOLE/PARTS schema it is essentially the isomorphic to the segment problem in the earlier Figure 12. An interesting question is to what degree do students conceptualize these problems in terms of the general schema and how much success do they enjoy in their problem-solving if they do.

The protocols that we obtained from the six students who were studying a regular course in geometry are relevant here. One of the sessions occurred soon after students had completed work on proofs of the kind shown in Figure 12 involving line segments. Students had been introduced to a concept of additive combination of angles, called the angle-addition property, but they had not yet worked on proof problems of the kind shown in Figure 13, which was the first problem presented in the interview session.

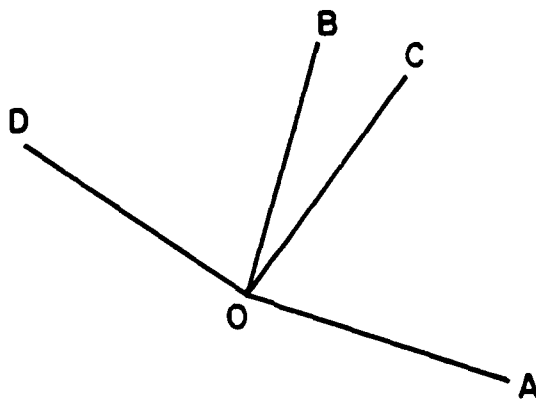
We found a full range of performance from these students in terms of their ability to recognize the similarity between the angle and segment problems. Four of the six noticed some similarity, one (who successfully solved the problem) noticed no similarity, and the final student had so many gaps in knowledge that it is difficult to diagnose exactly what this student knew. At least two students gave evidence of quite deep appreciation of the underlying similarity. One of them spontaneously



Given: \overline{RONY}
 $RO = NY$
Prove: $RN = OY$

FIGURE 12

Problem for learning a new schema.



Given: $\angle AOB$ and $\angle COD$ are right angles
Prove: $m\angle AOC = m\angle BOD$

FIGURE 13

Transfer problem.

remarked "Okay, well, it's basically the same kind of thing . . . Well, it's just working kind of the same thing. Except over a curve, kind of, the measure of angles instead of distances." The other remarked "Well, its the same problem again. You know something? I'm getting kind of tired of that problem." Although only the second of these two solved the problem, both showed evidence of trying to use the general conceptual structure on their problem-solving. We conclude from these protocols and others that at least some students make active use of these general schemata in their geometry problem-solving, that these schemata can be helpful in problem-solving, but that they are not essential. On the last point, our subject R whom we have studied most extensively, made slow but sure progress stubbornly refusing to try to relate the geometry material to prior knowledge.

Knowledge Compilation

In this section we discuss learning mechanisms that accomplish a transition from declarative to procedural knowledge. In the simulations of geometry learning that we have discussed in this chapter, we have assumed that the initial cognitive encoding of information for problem solving is declarative. There is room for disagreement on the question of whether problem-solving knowledge always has an initial declarative encoding (in fact the authors of this chapter disagree on this question -- see footnote 1). Even so, descriptions of procedures constitute an important source of information for human learners, and the development of cognitive procedures from initial knowledge in declarative form is an important requirement of much human learning.

One advantage of declarative representations over corresponding representations in the form of productions is that declarative representations are more concise. The same facts can give rise to a great many possible productions reflecting various ways that the information can be used. For instance, consider the textbook definition of supplementary angles:

"Supplementary angles are two angles whose measures have sum 180."

Below are productions that embody just some of the ways in which this knowledge can be used. These productions differ in terms of whether one is reasoning forward or backward, what the current goal is, and what is known.

IF $m\angle A + m\angle B = 180^\circ$
THEN $\angle A$ and $\angle B$ are supplementary

IF the goal is to prove $\angle A$ and $\angle B$ are supplementary

THEN set as a subgoal to prove $m\angle A + m\angle B = 180^\circ$

IF $\angle A$ and $\angle B$ are supplementary
 THEN $m\angle A + m\angle B = 180^\circ$

IF $\angle A$ and $\angle B$ are supplementary
 and $m\angle A = X$
 THEN $m\angle B = 180^\circ - X$

IF $\angle A$ and $\angle B$ are supplementary
 and the goal is to find $m\angle A$
 THEN set as a subgoal to find $m\angle B$

IF the goal is to show $\angle A \cong \angle B$
 and $\angle A$ is supplementary to $\angle C$
 and $\angle B$ is supplementary to $\angle D$
 THEN set as a subgoal to prove $\angle C \cong \angle D$

A basic point is that the definition of supplementary angles is fundamentally declarative in the sense that it can be used in multiple ways and does not contain a commitment to how it will be used. It is unreasonable to suppose that, in encoding the definition, the system anticipates all the ways in which it might be used and creates a procedural structure for each.

A related difficulty has to do with encoding control information into working-backward productions. The actual implementations of a working-backward production require rather intricate knowledge and use of goal settings that might exceed the knowledge or information -- handling capacities of many learners.

Interpretive Application of Knowledge

Rather than assuming students directly encode such information into procedures we have assumed that they first encode this information declaratively. In the ACT system encoding information declaratively amounts to growing new semantic network structure to encode the information. We suppose general *interpretative* procedures then use this information according to the features of the particular circumstance. In earlier sections we discussed how this declarative knowledge was encoded in schema-like format and how some interpretive procedures would use this knowledge. As we will now describe, when declarative knowledge is used many times in a particular way, automatic learning processes in ACT will create new procedures that directly apply the knowledge without the interpretative step. We refer to this kind of learning as *knowledge compilation*.

In individual subjects we see a gradual shift in performance which we would like to put into correspondence with this compilation from the interpretative application of declarative knowledge to direct application of procedures. For instance, after reading a particular postulate students' applications of that postulate are both slow and halting. Students will often recite to themselves the postulate before trying to apply it -- or even go back and reread it. It seems that they need to activate the declarative representation in their working memory so that interpretative procedures can apply to the data of this representation. They typically match separately fragments of the postulate to the problem. We will see that such fragmentary application is typical of a general knowledge interpreter applying to a declarative representation. With repeated use, however, application of the postulate smooths out. It is no longer explicitly recalled and it is no longer possible as observer or subject to discriminate separate steps in the application of the procedure.

We will use the side-angle-side schema of Table 1 to discuss how the student switches from the initial piecemeal interpretive application of knowledge to direct, unitary procedures. For convenience that table is reproduced as Table 6. The following are some of the productions that are used to apply the schema knowledge in working backwards mode:

P1: IF the goal is to prove a statement
 and there is a schema that has this statement as conclusion
 THEN set as subgoals to match the background of the schema
 and after that to prove the hypothesis of the schema

This production recognizes that the schema is relevant to proving the problem. It would invoke the SAS schema in situations where the goal was to prove two triangles congruent. The next production to apply is:

P2: IF the goal is to match a set of statements
 THEN match the first statement in the set

Production P1 had set the subgoal of matching the statements in the background. This production above starts that process going by focusing on the first statement in the background. This production is followed by a production which iterates through the statements of the background.

P3: IF the goal is to match a statement in a set
 and the problem contains a match to the statement
 THEN go on to match the next statement in the set

(Actually, there is a call to a subroutine of productions which execute the matches to each statement.

Table 6

SAS Schema

Background

s1 is a side of $\triangle XYZ$
s2 is a side of $\triangle XYZ$
A1 is an angle of $\triangle XYZ$
A1 is included by s1 and s2
s3 is a side of $\triangle UVW$
s4 is a side of $\triangle UVW$
A2 is an angle of $\triangle UVW$
A2 is included by s3 and s4

Hypothesis

s1 is congruent to s3
s2 is congruent to s4
A1 is congruent to A2

Conclusion

$\triangle XYZ$ is congruent to $\triangle UVW$

Comment

This is the side-angle-side postulate

See Neves & Anderson, 1981). After all statements in the background have been matched, the following production sets the goal to prove the hypotheses:

P4: IF the goal is to match a set of statements
and the last statement in that set has been matched
THEN go on to the goal that follows

Composition

There are two major processes in knowledge compilation -- *composition* and *proceduralization*. When a series of productions apply in a fixed order, composition will create a new production that accomplishes the effect of the sequence in a single step (see Neves & Anderson). Composition, operating on the sequence of P1, P2, and P3, applied to the SAS schema, would put forth the production:

P5: IF the goal is to prove a statement
and there is a schema that has this statement as conclusion
and the schema has a statement as the first member of its background
and the problem contains a match to the statement
THEN set as subgoals to match the background
and within this subgoal to match the next statement of the background
and after that to prove the hypotheses of the schema

This production only applies in the circumstance that the sequence P1, P2, and P3 applied and this production will have the same effect in terms of changes to the data base. The details underlying composition are discussed in Neves and Anderson, but the gist of the process is easy to describe. The composed production collects in its condition all those clauses from the individual productions' conditions except those that are the product of the actions of earlier productions in the sequence. As an example of this exception P2 has in its condition that the goal is to match the set of statements. Since this goal was set by P1, earlier in the sequence, it is not mentioned in the condition of the composed production P5. Thus, the condition is a test of whether the circumstances are right for the full sequence of productions to execute. The action of the composed production collects all actions of the individual productions except those involved in setting transitory goals that are finished with by the end of the sequence. As an example of this exception, P2 sets the subgoal of matching the first statement of the background but P3 meets this subgoal. Therefore, the subgoal is not mentioned in the action of the composed production P5.

This composition process can apply to the product of earlier compositions. Although there is

nothing special about compositions of three, consider what the resulting production would be like if P5 were composed with two successive iterative applications of P3:

P6: IF the goal is to prove a statement
 and there is a schema that has this statement as conclusion
 and the schema has a statement as the first member of the background
 and the problem contains a match to this statement
 and the schema has another statement as the next member of its background
 and the problem contains a match to this statement
 and the schema has another statement as the next member of its background
 and the problem contains a match to this statement
 THEN set as subgoals to match the background
 and within this the next statement of the background
 and after that to prove the hypotheses of the schema

It should be noted that such productions are not really specific to the SAS schema. Indeed, productions such as P5 and P6 might already have been formed from compositions derived from the productions applying to other, earlier schemata. If so, these composed productions would be ready to apply to the current schema. Thus, there can be some general transfer of learning produced by composition. However, there is a clear limit on how large such composed productions can become. As they get larger they require more information in the schema be retrieved from long term memory and held active in working memory. Limits on the capacity of working memory imply limits on the size of the general, interpretive conditions that can successfully match.

Proceduralization

Proceduralization is a process that eliminates retrieval of information from long term memory by creating productions with the knowledge formerly retrieved from long-term memory built into them. To illustrate the process of proceduralization, consider its application to the production P6. This statement contains in its condition four clauses that require retrieval of information from long term memory:

1. There is a schema that has the to-be-proven statement as its conclusion.
2. The schema has a statement as the first member of its background.
3. The schema has another statement as the next member of its background.
4. The schema has another statement as the next member of its background.

Applied to the SAS schema these statements match the following information:

1. The SAS schema has as its conclusion " $\triangle XYZ \cong \triangle UVW$ ".
2. The first statement of its background is "S1 is a side of $\triangle XYZ$ ".
3. The next statement of its background is "S2 is a side of $\triangle XYZ$ ".
4. The next statement of its background is "A1 is an angle of $\triangle XYZ$ ".

What is accomplished by matching these statements in P6 is to identify the SAS schema, its conclusion, and the first three statements of its background. A specialized production can be built which contains this information and does not require the long term memory retrievals:

P7: IF the goal is to prove that $\triangle XYZ \cong \triangle UVW$
 and S1 is a side of $\triangle XYZ$
 and S2 is a side of $\triangle XYZ$
 and A1 is an angle of $\triangle XYZ$
 THEN set as subgoals to match the background of the SAS schema
 and within this to match the next statement in the schema
 and after that to prove the hypothesis of the schema

This production is now specialized to the SAS schema and does not require any long term memory retrieval. Rather, built into its condition are the patterns retrieved from long term memory.

The effect of this proceduralization process is to enable larger composed productions to apply because the proceduralized productions are not limited by the need to retrieve long-term information into working memory. This in turn allows still larger compositions to be formed. The eventual product of the composition process applied to the top-down evocation of the SAS schema, initially via productions P1, P2, P3, and P4, would be:

P8: IF the goal is to prove that $\triangle XYZ$ is congruent to $\triangle UVW$
 and S1 is a side of $\triangle XYZ$
 and S2 is a side of $\triangle XYZ$
 and A1 is an angle of $\triangle XYZ$
 and A1 is included by S1 and S2
 and S3 is a side of $\triangle UVW$
 and S4 is a side of $\triangle UVW$
 and A2 is an angle of $\triangle UVW$
 and A2 is included by S3 and S4
 THEN set as subgoals to prove
 S1 is congruent to S3
 S2 is congruent to S4
 A1 is congruent to A2

This production serves to apply the SAS postulate in working backward mode. When the knowledge

reaches this state it has been completely proceduralized.

As we will discuss in later portions of the paper, composition need not stop when the postulate has been completely incorporated into a single production. Composition can continue to merge productions to compress even longer sequences of actions into a single production. For instance, consider what would happen should production P8 compose with later productions that attempted to prove the hypothesis parts. Suppose, furthermore, that the first two parts of the hypothesis could be established directly since they were already true. The composition process would produce the following working backward production:

P9: IF the goal is to prove that $\triangle XYZ$ is congruent to $\triangle UVW$
 and S1 is a side of $\triangle XYZ$
 and S2 is a side of $\triangle XYZ$
 and A1 is an angle of $\triangle XYZ$
 and A1 is included by S1 and S2
 and S3 is a side of $\triangle UVW$
 and S4 is a side of $\triangle UVW$
 and A2 is an angle of $\triangle UVW$
 and A2 is included by S3 and S4
 and S1 is congruent to S3
 and S2 is congruent to S4
 THEN set as a subgoal to prove that A1 is congruent to A2

This production checks that two sides of the triangles are congruent and sets the goal to prove that the included angles are congruent. P9 is obviously much more discriminant in its application than P8 and is therefore much more likely to lead to success.

Evidence for Composition and Proceduralization

So far we have offered two lines of argument that there are these processes of composition and proceduralization. One is that it creates a sensible connection between declarative knowledge and procedural knowledge. That is, knowledge starts out in a declarative form so that it can be used in multiple ways. However, if the knowledge is repeatedly used in the same way, efficient procedures will be created to apply the knowledge in that way. The second argument for these processes is that they are consistent with the gross qualitative features of the way application of knowledge smooths out and speeds up. That is, with practice explicit verbal recall of the geometry statements drop out and the piecemeal application becomes more unitary.

The idea is a natural one, that skill develops by collapsing together multiple steps in one. Lewis

(1976), who introduced composition applied to productions, traces the general concept back to Book (1908). However, there is more than intuitive appeal and general plausibility going for this learning mechanism. It is capable of accounting for a number of important facts about skill development. One feature of the knowledge compilation is that procedures can develop to apply the knowledge in one manner without corresponding procedures developing to apply the knowledge in other ways. It is somewhat notorious that people's ability to use knowledge can be specific to how the knowledge is evoked. For instance, Greeno and Magone (Note 1) have found that students who have a fair facility at proof generation make gross errors at proof checking, a skill which they have not practiced.

Neves and Anderson (this volume) provide an extensive discussion of how composition and proceduralization serves to account for a range of results in the experimental literature -- for the speed-up of a skill with practice, for the growing automaticity of a skill, for the Einstellung effect (Luchins, 1942), and for the drop-out of self-reports with practice.

Knowledge Optimization

Having operators proceduralized is not enough to guarantee successful proof generation. There is still a potentially very large search space of forward and backward inferences. Finding the proof tree in this search space would often be infeasible without some search heuristics that enable the system to try the right inferences first.

In our observations of student subjects as they learned geometry, we saw very little success in discovering such heuristics. Therefore, these observations do not provide a strong basis for the assertion that acquisition of such heuristics is an important part of learning geometry. However, the performance of students at more advanced stages of learning included many examples of problem-solving activity organized according to quite strong heuristic methods. An additional basis for our beliefs on this matter come from comparing our own performance on proof problems with that of that subjects whom we observed in their early stages of learning. While the title "expert" is a little overblown in our case, we have something of a novice-expert contrast here. Our two beginning subjects barely managed to get their knowledge beyond the initial proceduralization and often made choices in search that seem transparently wrong to us. Presumably, our more tuned judgment

reflects the acquisition of appropriate heuristics with experience. Therefore, in discussing particular heuristics we will be drawing on (a) those rare instances of learning of heuristics identifiable in our beginning subjects, (b) performance of more advanced students that has been analyzed previously, and (c) our own intuitions about the kinds of heuristics we use.

One kind of heuristic amounts to adding some discriminative conditions to a production to restrict its applicability. For instance, production P9 differs from P8 (pp. 24 & 25) by the addition of tests for two out of three of the conditions of SAS. While satisfying these conditions does not guarantee that SAS will be satisfied, it does make it more likely. This is the nature of a heuristic -- to select an operator on the basis of tests that suggest that it has a higher probability of success in this circumstance than other operators.

It is interesting to note that novices do not deal with proofs by plunging into endless search. They are very restrictive in what paths they attempt and are quite unwilling to consider all the paths that are legally possible. The problem is, of course, that the paths they select are often non-optimal or just plain dead-ends. Thus, at a general level, expertise does not develop by simply becoming more restrictive in search, rather it develops by becoming more appropriately restrictive.

There are four ways that we have been able to discover by which subjects can learn to make better choices in searching for a proof tree. One is by analogy to prior problems -- using with the current problem methods that succeeded in similar past problems. Use of such analogy is limited in ways that we discussed earlier and we will not discuss it further here. The second, related technique is to generalize from specific problems operators that capture what the solutions to these specific problems have in common. The third is a discrimination process by which restrictions are added to the applicability of more general operators. These restrictions are derived from a comparison of where the general operators succeeded and failed. The fourth process is a composition process by which sequences of operators become collapsed into single operators that apply in more restrictive situations. We will discuss the last three of these methods.

Generalization

We have characterized solving problems by analogy as superficial. Part of what is superficial about the approach is that the analogy is based only on the statement of the problems not on the structures of their solution. Analogy, in the sense discussed, cannot use the structure of the solution, because the proof for the second problem is not available when the analogy has to be made. Analogy is being used in service of finding the second proof.

Generalization, on the other hand, is based on a comparison between two problems and their solutions. By using the structure of the solution it is possible to select out the relevant aspects of the problem statement. A rule is formulated by the generalization process which tries to formulate what the two problems and their solutions have in common. That rule can then be used should similar problems appear. For instance, consider the two problems in Figure 14. The generalization process applied to these two examples would encode what they have in common by the following schema:

GENERALIZED SCHEMA:

Background

$\triangle XYZ$ contains $\triangle SYZ$
 $\triangle UVW$ contains $\triangle TVW$

Givens

$\overline{SY} \cong \overline{TV}$
 $\angle YSZ \cong \angle VTW$

Goal

$\triangle XYZ \cong \triangle UVW$

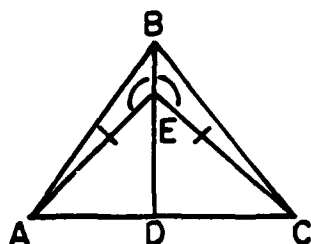
Method

$\triangle SYZ \cong \triangle TVW$ by SAS
 $\overline{YZ} \cong \overline{VW}$ by corresponding parts
 $\angle XYZ \cong \angle UVW$ by corresponding parts
 $\triangle XYZ \cong \triangle UVW$ by SAS

In our opinion, these generalizations are based on the same partial-matching process that underlies analogy. However, the partial-matching occurs between solved problems not just between problem statements. Because the product of the partial match is a fairly general problem description, it is likely to apply to many problems. Thus it is likely to be strengthened and become a permanent part of the student's repertoire for searching for proofs. This contrasts to the specific examples that serve as the basis for analogy. These specific examples are likely to be forgotten.

We have been able to identify two moderately clear cases of generalization in R's protocols. One

(a)

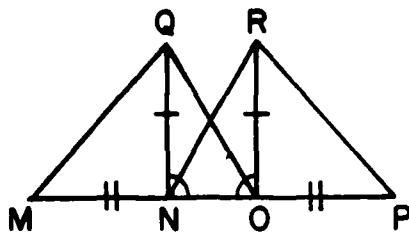


Given: $\overline{AE} \cong \overline{EC}$

$\angle BEA \cong \angle BEC$

Prove: $\triangle ABD \cong \triangle CBD$

(b)



Given: $\overline{QN} \cong \overline{OR}$

$\angle QON \cong \angle RON$

$\overline{MN} \cong \overline{OP}$

Prove: $\triangle MQO \cong \triangle PRN$

FIGURE 14

Two proof problems whose generalization leads to a useful operator.

has to do with problems of the variety illustrated in part (a) of Figure 6. Many variations on this problem appeared in the early part of the text and R came to recognize this general type of problem when it appeared later. The other example has to do with the use of the hypotenuse-leg theorem for right angle triangles. After some examples R formulated the generalized rule that he should use this theorem if he was presented with two right angle triangles whose hypotenuses were given as congruent. This is the feature that all the hypotenuse-log problems had in common.

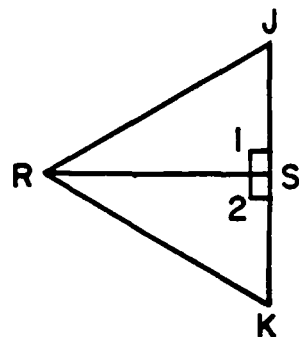
Discrimination

Discrimination provides a complementary process to generalization. It takes operators that are too general and thus are applying in incorrect situations and places restrictions on their range of applicability. If the operator to be discriminated is embodied as a production, discrimination adds an additional clause to restrict the range of situations where the production condition will match. ACT determines what additional clauses to add by comparing the difference between successful and unsuccessful application of the rule.

Figure 15 illustrates an analysis of a problem which led subject R to form a discrimination. In part (a) we have a representation of the problem and in part (b) we have indicated the network of backward inferences that constitute R's attempt to solve the problem. First he tried to use SSS, a method which had worked on a previous problem that had a great deal of superficial similarity to this problem. However, he was not able to get the sides \overline{RK} and \overline{RS} congruent. Then he switched to SAS, the only other method he had at the time for proving triangles congruent. Interestingly, it was only in the context of this goal that he recognized the right angles were congruent. After he had finished with this problem, he verbally announced the rule to use SSS only if there was no angle mentioned. This can be seen to be the product of discrimination. The "don't use SSS if angle" comes from a comparison of the previous problem in which no angle was mentioned with the current problem that did mention angles.

ACT's generalization and discrimination processes were described in considerable detail in Anderson, Kline, and Beasley (1979). There we were concerned with showing how they applied in modelling the acquisition of category schema or prototypes. That data provided pretty strong evidence for the ACT mechanisms and further new data is contained in Elio and Anderson

(a)



Given: $\angle 1$ and $\angle 2$ are
right angles
 $\overline{JS} \cong \overline{KS}$

Prove: $\triangle RSJ \cong \triangle RSK$

(b)

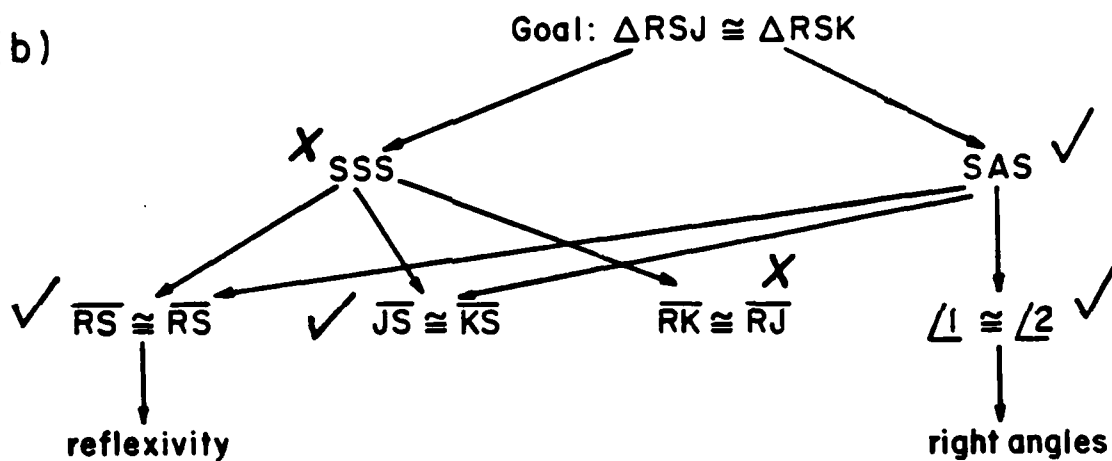


FIGURE 15

Problem leading to a discrimination.

(forthcoming). It is partly the success of that enterprise that leads us to believe they play an important role in the development of expertise in geometry proof generation. Basically, the claim is that students develop from examples schemata for when various proof methods are appropriate just as they develop schemata for what are examples of categories.

Composition

We feel that composition has an important role to play in forming multiple operator sequences just as it played an important role in the initial proceduralization of operators. Figure 16 illustrates an example where composition can apply. The first production to apply in solving this problem would be:

P1: IF the goal is to prove $\angle X \cong \angle U$
 and $\angle X$ is part of $\triangle XYZ$
 and $\angle U$ is part of $\triangle UVW$
 THEN the subgoal is to prove $\triangle XYZ \cong \triangle UVW$

This production would set as a subgoal to prove $\triangle ABC \cong \triangle DBC$. At this point the following production might apply:

P2: IF the goal is to prove $\triangle XYZ \cong \triangle UVW$
 and $\overline{XY} \cong \overline{UV}$
 and $\overline{ZX} \cong \overline{WU}$
 THEN the subgoal is to prove $\overline{YZ} \cong \overline{VW}$

This production, applied to the situation in Figure 16, would set as the subgoal to prove $\overline{BC} \cong \overline{BC}$ as a step on the way to using SSS. At this point the following production would apply:

P3: IF the goal is to prove $\overline{XY} \cong \overline{XY}$
 THEN this may be concluded by reflexivity

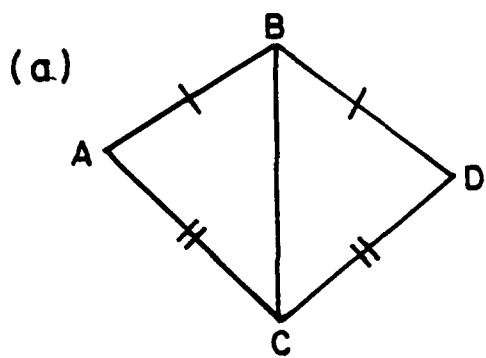
This production would add $\overline{BC} \cong \overline{BC}$ and allow the following production to apply:

P4: IF the goal is to prove $\triangle XYZ \cong \triangle UVW$
 and $\overline{XY} \cong \overline{UV}$
 and $\overline{YZ} \cong \overline{VW}$
 and $\overline{ZX} \cong \overline{WU}$
 THEN the goal may be concluded by SSS

where $\overline{XY} = \overline{AB}$, $\overline{UV} = \overline{DB}$, $\overline{YZ} = \overline{BC}$, $\overline{VW} = \overline{BC}$, $\overline{ZX} = \overline{CA}$, and $\overline{WU} = \overline{CD}$. This adds the information that $\triangle ABC \cong \triangle DBC$. Finally, the following production will apply which recognizes that the to-be-proven conclusion is now established:

P5: IF the goal to prove $\angle X \cong \angle U$
 and $\triangle XYZ \cong \triangle UVW$
 THEN the goal may be concluded because of congruent parts of congruent triangles

The composition process operating on this sequence of productions, would eventually produce a



Given: $\overline{AB} \cong \overline{DB}$
 $\overline{CA} \cong \overline{CD}$
Prove: $\angle A \cong \angle D$

FIGURE 16

Problem leading to a composition.

production of the form:

P6: IF the goal is to prove $\angle A \cong \angle D$
 and $\angle A$ is part of $\triangle ABC$
 and $\angle D$ is part of $\triangle DBC$
 and $\overline{AB} \cong \overline{DB}$
 and $\overline{CA} \cong \overline{DC}$
 THEN conclude $\overline{AB} \cong \overline{AB}$ by reflexivity
 and conclude $\triangle ABC \cong \triangle DBC$ by SSS
 and conclude the goal because of congruent parts of congruent triangles

The variables in this production have been named to correspond to the terms in Figure 16 for purposes of readability. This production would immediately recognize the solution to a problem like that in Figure 16. Another feature of composition, illustrated in this example, is that it transforms what had been a basically working backward solution to the problem into something much more of the character of working forward. Indeed, all the methods that we have discussed for tuning search operators, to the extent that they put into the conditions additional tests for applicability and into the action additional inferences, have the effect of converting working backward into working forward. Larkin, McDermott, Simon, and Simon (in press) and Larkin (this conference) have commented on this same transformation in the character of physics problem solving with the development of expertise.

Summary

By way of a summary, Figure 17 provides an overview of the progress that we think a student makes through geometry learning. There are two major sources of knowledge in the learning environment -- these are the general rules stated in the instructional portion of the text and the examples of proofs worked out and provided as exercises. Both rules and examples are given declarative encodings. The declarative representation of rules can be used to solve problems by general problem-solving procedures. The declarative encoding of the examples can be used to guide the solution to problems through a declarative interpreter. The twin processes of knowledge compilation, composition and proceduralization, can transform either of these declarative representations into a procedural form. The procedures compiled from the declarative representation of the rules are general and the procedures compiled from examples are specific. The generalization mechanism provides a way of transforming the specific procedures into general form.

There is another route to learning and this is the process of subsumption which involves the

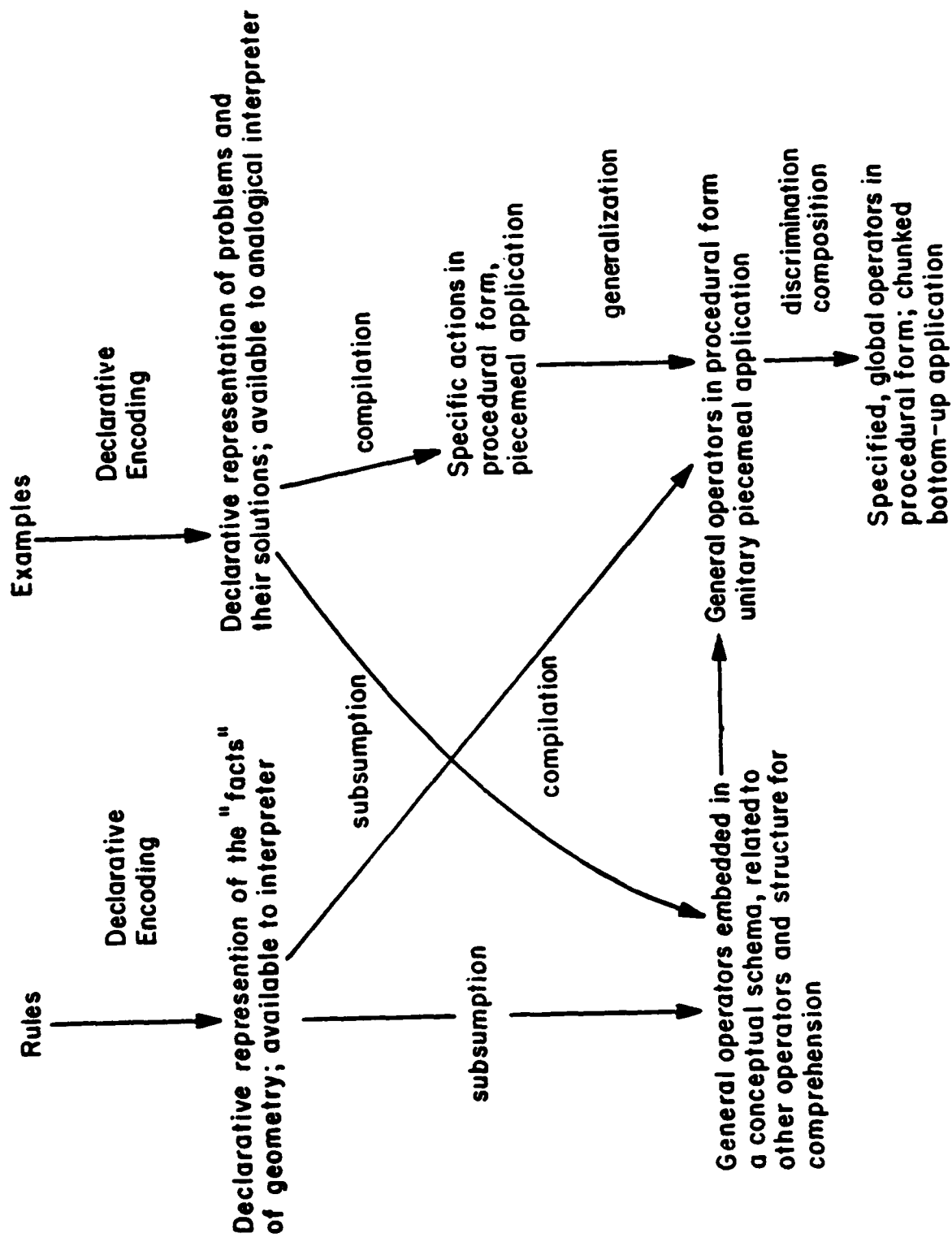


FIGURE 17

Summary of Learning Processes.

development of new problem-solving schemata out of old ones. We see this as a form of learning with structural understanding. We discussed two types of learning of this variety -- elaborating existing schemata to apply to new situations or building new schemata out of existing schemata. These schemata are basically declarative in character and the compilation process should be able to apply to these to form general procedures also.

Note that there are many routes by which a student can arrive at general operators for solving geometry problems. This corresponds to the diversity we see in individual students: some lean heavily on prior knowledge in learning; others try to apply the general rules of geometry directly; still others (probably the majority) lean heavily on past examples to guide their problem solving and learning.

Whatever means the student takes to achieve general operators for solving geometry problems, there remains a great deal of learning about heuristic features of the problem environment that are predictive of solution steps. To some extent, general operators that arise from specific examples through generalization may still preserve some of these heuristic features of the examples. Other methods of acquiring these heuristic features involve the processes of discrimination and composition which create larger multiple-inference operators which are much more discriminant in their range of applicability. In the extreme we get special rules that outline full proof trees for certain kinds of problems. The character of these operators is, as we have noted, working forward more than working backward. To the extent that new problems fit the specifications of these advanced operators, solution will be quick and efficient. However, to the extent new problems pose novel configurations of features not covered by the advanced operators the student will have to fall back to the slower and more general operators for working backwards. We certainly notice this variation in our own behavior as "experts" depending on how unique a geometry problem is. Our view of expertise, then, is very much like the one that was developed for chess (Chase & Simon, 1973; Simon & Gilmarin, 1973); that is, experts in geometry proof generation have simply encoded many special case rules.

In conclusion, we should make some remarks to avoid an overly-impressive interpretation of Figure 17: Nowhere does there reside a single version of the ACT system containing all the components in

Figure 17 which progresses from the initial input of a geometry text to the final status of a geometry expert. Rather we have simulated bits and pieces of Figure 17 separately. All the major components have been tested on problems and we would like to believe they would all work together if put into a complete system. However, undoubtedly interesting new issues would come up if we tried to put it all together. So, Figure 17 represents a partial sketch of what is involved in geometry learning.

There are two basic reasons for the current disjointed character of our simulations. One is that the size and complexity of the full system is staggering. We run out of PDP 10 conventional address space just simulating components. The other reason is that we all took on separate parts of the task, and while we certainly talked to each other, our implementation efforts were separate. While on this topic we should indicate who was mainly responsible for implementing which components: John Anderson implemented the general control structure for proof searches; Jim Greeno the processes underlying schema subsumption; Paul Kline the analogy components; David Neves the knowledge compilation processes; and John Anderson and Paul Kline the processes of generalization and discrimination. Perhaps someday, technology and ourselves willing, there will be implemented an integrated version of Figure 17.

Reference Notes

1. Greeno, J.G. & Magone, M.E. *The concept of formal proof*. Manuscript in preparation.
2. Heller, J.I. *The role of "focus" in children's understanding of arithmetic word problems*. Paper presented at the meeting of the American Educational Research Association, Boston, April 1980.
3. Neches, R. *Promoting self-discovery of improved strategies*. Paper presented at the meeting of the American Educational Research Association, San Francisco, March 1978.
4. Riley, M.S. *The development of children's ability to solve arithmetic word problems*. Paper presented at the meeting of the American Educational Research Association, San Francisco, April 1979.
5. Steele, G.L., Jr., & Sussman, G.J. *Constraints* (AI Memo No. 502). Cambridge, Mass.: Massachusetts Institute of Technology, Artificial Intelligence Laboratory, November 1978.

References

- Anderson, J.R. *Language, Memory, and Thought*. Hillsdale, N.J.: Lawrence Erlbaum Associates, 1976.
- Anderson, J.R., Kline, P.J., and Beasley, C.M. A general learning theory and its application to schema abstraction. In G.H. Bower (Ed.), *The Psychology of Learning and Motivation*, Vol. 13, New York: Academic Press, 1979.
- Anderson, J.R., Kline, P.J., and Beasley, C.M. Complex learning processes. In R.E. Snow, P.A. Federico, and W.E. Montague (Eds.), *Aptitude, Learning, and Instruction: Cognitive Process Analyses*. Hillsdale, N.J.: Lawrence Erlbaum Associates, 1980.
- Bobrow, D.G. and Winograd, T. An overview of KRL, a knowledge representation language. *Cognitive Science*, 1977, 1, 3-46.
- Book, W.F. *The psychology of skill with special reference to its acquisition in typewriting*. Missoula, Montana: University of Montana, 1908. Facsimile in *The Psychology of Skill*, New York: Arno Press, 1973.
- Chase, W.G. and Simon, H.A. The mind's eye in chess. In W.G. Chase (Ed.), *Visual Information Processing*. New York: Academic Press, 1973.
- Elio, R. and Anderson, J.R. Effects of category generalizations and instance similarity on schema abstraction, in preparation.
- Gick, M.L. and Holyoak, K.J. Analogical problem solving. *Cognitive Psychology*, 1980, 12, 306-355.
- Greeno, J.G. Indefinite goals in well-structured problems. *Psychological Review*, 1976, 83, 479-491.
- Greeno, J.G. A study of problem solving. In R. Glaser (Ed.), *Advances in instructional psychology* (Vol. 1). Hillsdale, N.J.: Lawrence Erlbaum Associates, 1978.
- Greeno, J.G., Magone, M.E., and Chaiklin, S. Theory of constructions and set in problem solving. *Memory and Cognition*, 1979, 7, 445-461.
- Jurgensen, R.C., Donnelly, A.J., Maier, J.E. and Rising, G.R. *Geometry*. Boston, MA: Houghton Mifflin Co., 1975.
- Katona, G. *Organizing and memorizing*. New York: Columbia University Press, 1940.
- Larkin, J.H., McDermott, J., Simon, D.P., and Simon, H.A. Models of competence in solving physics problems. *Cognitive Science*, 1980, in press.
- Lewis, C.H. Production system models of practice effects. Unpublished Ph.D. dissertation, University of Michigan, 1978.
- Luchins, A.S. Mechanization in problem solving. *Psychological Monographs*,

1942, 54, No. 248.

Newell, A. and Simon, H. *Human problem solving*. Englewood Cliffs, N.J.: Prentice-Hall, 1972.

Sacerdoti, E.D. *A structure for plans and behavior*. New York: Elsevier North-Holland, 1977.

Simon, H.A. and Gilmarin, K. A simulation of memory for chess positions. *Cognitive Psychology*, 1973, 5, 29-46.

Wertheimer, M. *Productive thinking*. New York: Harper & Row, 1945. (Enlarged edition, 1959)

Winston, P.H. Learning by creating transfer frames. *Artificial Intelligence*, 1978, 10, 147-172.

Navy

- 1 Dr. Ed Aiken
Navy Personnel R&D Center
San Diego, CA 92152
- 1 Dr. Robert Breaux
Code N-711
NAVTRAEQUIPCEN
Orlando, FL 32813
- 1 Dr. Richard Elster
Department of Administrative Sciences
Naval Postgraduate School
Monterey, CA 93940
- 1 DR. PAT FEDERICO
NAVY PERSONNEL R&D CENTER
SAN DIEGO, CA 92152
- 1 Dr. John Ford
Navy Personnel R&D Center
San Diego, CA 92152
- 1 Dr. Henry M. Halff
Department of Psychology, C-009
University of California at San Diego
La Jolla, CA 92093
- 1 LT Steven D. Harris, MSC, USN
Code 6021
Naval Air Development Center
Warminster, Pennsylvania 18974
- 1 CDR Charles W. Hutchins
Naval Air Systems Command Hq
AIR-340F
Navy Department
Washington, DC 20361
- 1 CDR Robert S. Kennedy
Head, Human Performance Sciences
Naval Aerospace Medical Research Lab
Box 29407
New Orleans, LA 70189
- 1 Dr. Norman J. Kerr
Chief of Naval Technical Training
Naval Air Station Memphis (75)
Millington, TN 38054

Navy

- 1 Dr. William L. Maloy
Principal Civilian Advisor for
Education and Training
Naval Training Command, Code 00A
Pensacola, FL 32508
- 1 Dr. Kneale Marshall
Scientific Advisor to DCNO(MPT)
OP01T
Washington DC 20370
- 1 CAPT Richard L. Martin, USN
Prospective Commanding Officer
USS Carl Vinson (CVN-70)
Newport News Shipbuilding and Drydock Co
Newport News, VA 23607
- 1 Dr William Montague
Navy Personnel R&D Center
San Diego, CA 92152
- 1 Library
Naval Health Research Center
P. O. Box 85122
San Diego, CA 92138
- 1 Naval Medical R&D Command
Code 44
National Naval Medical Center
Bethesda, MD 20014
- 1 Ted M. I. Yellen
Technical Information Office, Code 201
NAVY PERSONNEL R&D CENTER
SAN DIEGO, CA 92152
- 1 Library, Code P201L
Navy Personnel R&D Center
San Diego, CA 92152
- 1 Technical Director
Navy Personnel R&D Center
San Diego, CA 92152
- 6 Commanding Officer
Naval Research Laboratory
Code 2627
Washington, DC 20390

Navy

- 1 Psychologist
ONR Branch Office
Bldg 114, Section D
666 Summer Street
Boston, MA 02210
- 1 Psychologist
ONR Branch Office
536 S. Clark Street
Chicago, IL 60605
- 1 Office of Naval Research
Code 437
800 N. Quincy SStreet
Arlington, VA 22217
- 5 Personnel & Training Research Programs
(Code 458)
Office of Naval Research
Arlington, VA 22217
- 1 Psychologist
ONR Branch Office
1030 East Green Street
Pasadena, CA 91101
- 1 Special Asst. for Education and
Training (OP-01E)
Rm. 2705 Arlington Annex
Washington, DC 20370
- 1 Office of the Chief of Naval Operations
Research Development & Studies Branch
(OP-115)
Washington, DC 20350
- 1 Captain Donald F. Parker, USN
Commanding Officer
Navy Personnel R&D Center
San Diego, CA 92152
- 1 LT Frank C. Petho, MSC, USN (Ph.D)
Code L51
Naval Aerospace Medical Research Laborat
Pensacola, FL 32508

Navy

- 1 Dr. Gary Poock
Operations Research Department
Code 55PK
Naval Postgraduate School
Monterey, CA 93940
- 1 Roger W. Remington, Ph.D
Code L52
NAMRL
Pensacola, FL 32508
- 1 Dr. Bernard Rimland (03B)
Navy Personnel R&D Center
San Diego, CA 92152
- 1 Dr. Worth Scanland
Chief of Naval Education and Training
Code N-5
NAS, Pensacola, FL 32508
- 1 Dr. Robert G. Smith
Office of Chief of Naval Operations
OP-987H
Washington, DC 20350
- 1 Dr. Alfred F. Smode
Training Analysis & Evaluation Group
(TAEG)
Dept. of the Navy
Orlando, FL 32813
- 1 Dr. Richard Sorensen
Navy Personnel R&D Center
San Diego, CA 92152
- 1 Roger Weissinger-Baylon
Department of Administrative Sciences
Naval Postgraduate School
Monterey, CA 93940
- 1 Dr. Robert Wisher
Code 309
Navy Personnel R&D Center
San Diego, CA 92152

Navy

- 1 Mr John H. Wolfe
Code P310
U. S. Navy Personnel Research and
Development Center
San Diego, CA 92152

Army

- 1 Technical Director
U. S. Army Research Institute for the
Behavioral and Social Sciences
5001 Eisenhower Avenue
Alexandria, VA 22333
- 1 HQ USAREUE & 7th Army
ODCSOPS
USAREUE Director of GED
APO New York 09403
- 1 DR. RALPH DUSEK
U.S. ARMY RESEARCH INSTITUTE
5001 EISENHOWER AVENUE
ALEXANDRIA, VA 22333
- 1 DR. FRANK J. HARRIS
U.S. ARMY RESEARCH INSTITUTE
5001 EISENHOWER AVENUE
ALEXANDRIA, VA 22333
- 1 Col Frank Hart
Army Research Institute for the
Behavioral & Social Sciences
5001 Eisenhower Blvd.
Alexandria, VA 22333
- 1 Dr. Michael Kaplan
U.S. ARMY RESEARCH INSTITUTE
5001 EISENHOWER AVENUE
ALEXANDRIA, VA 22333
- 1 Dr. Milton S. Katz
Training Technical Area
U.S. Army Research Institute
5001 Eisenhower Avenue
Alexandria, VA 22333
- 1 Dr. Harold F. O'Neil, Jr.
Attn: PERI-OK
Army Research Institute
5001 Eisenhower Avenue
Alexandria, VA 22333
- 1 Dr. Robert Sasmor
U. S. Army Research Institute for the
Behavioral and Social Sciences
5001 Eisenhower Avenue
Alexandria, VA 22333

Army

- 1 Dr. Frederick Steinheiser
 U. S. Army Reserch Institute
 5001 Eisenhower Avenue
 Alexandria, VA 22333

- 1 Dr. Joseph Ward
 U.S. Army Research Institute
 5001 Eisenhower Avenue
 Alexandria, VA 22333

Air Force

- 1 Dr. Earl A. Alluisi
 HQ, AFHRL (AFSC)
 Brooks AFB, TX 78235

- 1 Dr. Genevieve Haddad
 Program Manager
 Life Sciences Directorate
 AFOSR
 Bolling AFB, DC 20332

- 1 Dr. Marty Rockway
 Technical Director
 AFHRL(OT)
 Williams AFB, AZ 58224

- 2 3700 TCHTW/TTGH Stop 32
 Sheppard AFB, TX 76311

- 1 Jack A. Thorp, Maj., USAF
 Life Sciences Directorate
 AFOSR
 Bolling AFB, DC 20332

Marines

- 1 H. William Greenup
Education Advisor (E031)
Education Center, MCDEC
Quantico, VA 22134
- 1 Special Assistant for Marine
Corps Matters
Code 100M
Office of Naval Research
800 N. Quincy St.
Arlington, VA 22217
- 1 DR. A.L. SLAFKOSKY
SCIENTIFIC ADVISOR (CODE RD-1)
HQ, U.S. MARINE CORPS
WASHINGTON, DC 20380

Other DoD

- 12 Defense Technical Information Center
Cameron Station, Bldg 5
Alexandria, VA 22314
Attn: TC
- 1 Dr. Craig I. Fields
Advanced Research Projects Agency
1400 Wilson Blvd.
Arlington, VA 22209
- 1 Dr. Dexter Fletcher
ADVANCED RESEARCH PROJECTS AGENCY
1400 WILSON BLVD.
ARLINGTON, VA 22209
- 1 Military Assistant for Training and
Personnel Technology
Office of the Under Secretary of Defense
for Research & Engineering
Room 3D129, The Pentagon
Washington, DC 20301

Civil Govt

- 1 Dr. Susan Chipman
Learning and Development
National Institute of Education
1200 19th Street NW
Washington, DC 20208
- 1 Dr. Joseph I. Lipson
SEDR W-638
National Science Foundation
Washington, DC 20550
- 1 William J. McLaurin
Rm. 301, Internal Revenue Service
2221 Jefferson Davis Highway
Arlington, VA 22202
- 1 Dr. Arthur Melmed
National Institute of Education
1200 19th Street NW
Washington, DC 20208
- 1 Dr. Andrew R. Molnar
Science Education Dev.
and Research
National Science Foundation
Washington, DC 20550
- 1 Personnel R&D Center
Office of Personnel Management
1900 E Street NW
Washington, DC 20415
- 1 Dr. Frank Withrow
U. S. Office of Education
400 Maryland Ave. SW
Washington, DC 20202
- 1 Dr. Joseph L. Young, Director
Memory & Cognitive Processes
National Science Foundation
Washington, DC 20550

Non Govt

- 1 Anderson, Thomas H., Ph.D.
Center for the Study of Reading
174 Children's Research Center
51 Gerty Drive
Champaign, IL 61820
- 1 Dr. John Annett
Department of Psychology
University of Warwick
Coventry CV4 7AL
ENGLAND
- 1 DR. MICHAEL ATWOOD
SCIENCE APPLICATIONS INSTITUTE
40 DENVER TECH. CENTER WEST
7935 E. PRENTICE AVENUE
ENGLEWOOD, CO 80110
- 1 1 psychological research unit
Dept. of Defense (Army Office)
Campbell Park Offices
Canberra ACT 2600, Australia
- 1 Dr. Alan Baddeley
Medical Research Council
Applied Psychology Unit
15 Chaucer Road
Cambridge CB2 2EF
ENGLAND
- 1 Dr. Patricia Baggett
Department of Psychology
University of Denver
University Park
Denver, CO 80208
- 1 Mr Avron Barr
Department of Computer Science
Stanford University
Stanford, CA 94305
- 1 Dr. Nicholas A. Bond
Dept. of Psychology
Sacramento State College
600 Jay Street
Sacramento, CA 95819

Non Govt

- 1 Dr. Lyle Bourne
Department of Psychology
University of Colorado
Boulder, CO 80309
- 1 Dr. John S. Brown
XEROX Palo Alto Research Center
3333 Coyote Road
Palo Alto, CA 94304
- 1 Dr. Bruce Buchanan
Department of Computer Science
Stanford University
Stanford, CA 94305
- 1 DR. C. VICTOR BUNDERSON
WICAT INC.
UNIVERSITY PLAZA, SUITE 10
1160 SO. STATE ST.
OREM, UT 84057
- 1 Dr. Pat Carpenter
Department of Psychology
Carnegie-Mellon University
Pittsburgh, PA 15213
- 1 Dr. John B. Carroll
Psychometric Lab
Univ. of No. Carolina
Davie Hall 013A
Chapel Hill, NC 27514
- 1 Charles Myers Library
Livingstone House
Livingstone Road
Stratford
London E15 2LJ
ENGLAND
- 1 Dr. William Chase
Department of Psychology
Carnegie Mellon University
Pittsburgh, PA 15213
- 1 Dr. Micheline Chi
Learning R & D Center
University of Pittsburgh
3939 O'Hara Street
Pittsburgh, PA 15213

Non Govt

- 1 Dr. William Clancey
Department of Computer Science
Stanford University
Stanford, CA 94305
- 1 Dr. Allan M. Collins
Bolt Beranek & Newman, Inc.
50 Moulton Street
Cambridge, Ma 02138
- 1 Dr. Lynn A. Cooper
LRDC
University of Pittsburgh
3939 O'Hara Street
Pittsburgh, PA 15213
- 1 Dr. Meredith P. Crawford
American Psychological Association
1200 17th Street, N.W.
Washington, DC 20036
- 1 Dr. Hubert Dreyfus
Department of Philosophy
University of California
Berkely, CA 94720
- 1 LCOL J. C. Eggenberger
DIRECTORATE OF PERSONNEL APPLIED RESEARC
NATIONAL DEFENCE HQ
101 COLONEL BY DRIVE
OTTAWA, CANADA K1A 0K2
- 1 Dr. Ed Feigenbaum
Department of Computer Science
Stanford University
Stanford, CA 94305
- 1 Dr. Richard L. Ferguson
The American College Testing Program
P.O. Box 168
Iowa City, IA 52240
- 1 Mr. Wallace Feurzeig
Bolt Beranek & Newman, Inc.
50 Moulton St.
Cambridge, MA 02138

Non Govt

- 1 Dr. Victor Fields
Dept. of Psychology
Montgomery College
Rockville, MD 20850
- 1 Dr. John R. Frederiksen
Bolt Beranek & Newman
50 Moulton Street
Cambridge, MA 02138
- 1 Dr. Alinda Friedman
Department of Psychology
University of Alberta
Edmonton, Alberta
CANADA T6G 2E9
- 1 Dr. R. Edward Geiselman
Department of Psychology
University of California
Los Angeles, CA 90024
- 1 DR. ROBERT GLASER
LRDC
UNIVERSITY OF PITTSBURGH
3939 O'HARA STREET
PITTSBURGH, PA 15213
- 1 Dr. Marvin D. Glock
217 Stone Hall
Cornell University
Ithaca, NY 14853
- 1 Dr. Daniel Gopher
Industrial & Management Engineering
Technion-Israel Institute of Technology
Haifa
ISRAEL
- 1 DR. JAMES G. GREENO
LRDC
UNIVERSITY OF PITTSBURGH
3939 O'HARA STREET
PITTSBURGH, PA 15213
- 1 Dr. Harold Hawkins
Department of Psychology
University of Oregon
Eugene OR 97403

Non Govt

- 1 Dr. Barbara Hayes-Roth
The Rand Corporation
1700 Main Street
Santa Monica, CA 90406
- 1 Dr. Frederick Hayes-Roth
The Rand Corporation
1700 Main Street
Santa Monica, CA 90406
- 1 Dr. James R. Hoffman
Department of Psychology
University of Delaware
Newark, DE 19711
- 1 Glenda Greenwald, Ed.
"Human Intelligence Newsletter"
P. O. Box 1163
Birmingham, MI 48012
- 1 Dr. Earl Hunt
Dept. of Psychology
University of Washington
Seattle, WA 98105
- 1 Dr. Steven W. Keele
Dept. of Psychology
University of Oregon
Eugene, OR 97403
- 1 Dr. Walter Kintsch
Department of Psychology
University of Colorado
Boulder, CO 80302
- 1 Dr. David Kieras
Department of Psychology
University of Arizona
Tuscon, AZ 85721
- 1 Dr. Kenneth A. Klivington
Program Officer
Alfred P. Sloan Foundation
630 Fifth Avenue
New York, NY 10111

Non Govt

- 1 Dr. Stephen Kosslyn
Harvard University
Department of Psychology
33 Kirkland Street
Cambridge, MA 02138
- 1 Mr. Marlin Kroger
1117 Via Goleta
Palos Verdes Estates, CA 90274
- 1 Dr. Jill Larkin
Department of Psychology
Carnegie Mellon University
Pittsburgh, PA 15213
- 1 Dr. Alan Lesgold
Learning R&D Center
University of Pittsburgh
Pittsburgh, PA 15260
- 1 Dr. Michael Levine
Department of Educational Psychology
210 Education Bldg.
University of Illinois
Champaign, IL 61801
- 1 Dr. Robert A. Levit
Director, Behavioral Sciences
The BDM Corporation
7915 Jones Branch Drive
McClean, VA 22101
- 1 Dr. Charles Lewis
Faculteit Sociale Wetenschappen
Rijksuniversiteit Groningen
Oude Boteringestraat
Groningen
NETHERLANDS
- 1 Dr. Erik McWilliams
Science Education Dev. and Research
National Science Foundation
Washington, DC 20550
- 1 Dr. Mark Miller
Computer Science Laboratory
Texas Instruments, Inc.
Mail Station 371, P.O. Box 225936
Dallas, TX 75265

Non Govt

- 1 Dr. Allen Munro
Behavioral Technology Laboratories
1845 Elena Ave., Fourth Floor
Redondo Beach, CA 90277
- 1 Dr. Donald A Norman
Dept. of Psychology C-009
Univ. of California, San Diego
La Jolla, CA 92093
- 1 Dr. Jesse Orlansky
Institute for Defense Analyses
400 Army Navy Drive
Arlington, VA 22202
- 1 Dr. Seymour A. Papert
Massachusetts Institute of Technology
Artificial Intelligence Lab
545 Technology Square
Cambridge, MA 02139
- 1 Dr. James A. Paulson
Portland State University
P.O. Box 751
Portland, OR 97207
- 1 MR. LUIGI PETRULLO
2431 N. EDGEWOOD STREET
ARLINGTON, VA 22207
- 1 DR. PETER POLSON
DEPT. OF PSYCHOLOGY
UNIVERSITY OF COLORADO
BOULDER, CO 80309
- 1 Dr. Steven E. Poltrock
Department of Psychology
University of Denver
Denver, CO 80208
- 1 MINRAT M. L. RAUCH
P II 4
BUNDESMINISTERIUM DER VERTEIDIGUNG
POSTFACH 1328
D-53 BONN 1, GERMANY

Non Govt

- 1 Dr. Fred Reif
SESAME
c/o Physics Department
University of California
Berkely, CA 94720
- 1 Dr. Andrew M. Rose
American Institutes for Research
1055 Thomas Jefferson St. NW
Washington, DC 20007
- 1 Dr. Ernst Z. Rothkopf
Bell Laboratories
600 Mountain Avenue
Murray Hill, NJ 07974
- 1 DR. WALTER SCHNEIDER
DEPT. OF PSYCHOLOGY
UNIVERSITY OF ILLINOIS
CHAMPAIGN, IL 61820
- 1 Dr. Alan Schoenfeld
Department of Mathematics
Hamilton College
Clinton, NY 13323
- 1 DR. ROBERT J. SEIDEL
INSTRUCTIONAL TECHNOLOGY GROUP
HUMRRO
300 N. WASHINGTON ST.
ALEXANDRIA, VA 22314
- 1 Committee on Cognitive Research
% Dr. Lonnie R. Sherrod
Social Science Research Council
605 Third Avenue
New York, NY 10016
- 1 Robert S. Siegler
Associate Professor
Carnegie-Mellon University
Department of Psychology
Schenley Park
Pittsburgh, PA 15213
- 1 Dr. Robert Smith
Department of Computer Science
Rutgers University
New Brunswick, NJ 08903

Non Govt

- 1 Dr. Richard Snow
School of Education
Stanford University
Stanford, CA 94305
- 1 Dr. Robert Sternberg
Dept. of Psychology
Yale University
Box 11A, Yale Station
New Haven, CT 06520
- 1 DR. ALBERT STEVENS
BOLT BERANEK & NEWMAN, INC.
50 MOULTON STREET
CAMBRIDGE, MA 02138
- 1 Dr. David Stone
ED 236
SUNY, Albany
Albany, NY 12222
- 1 DR. PATRICK SUPPES
INSTITUTE FOR MATHEMATICAL STUDIES IN
THE SOCIAL SCIENCES
STANFORD UNIVERSITY
STANFORD, CA 94305
- 1 Dr. Kikumi Tatsuoka
Computer Based Education Research
Laboratory
252 Engineering Research Laboratory
University of Illinois
Urbana, IL 61801
- 1 Dr. John Thomas
IBM Thomas J. Watson Research Center
P.O. Box 218
Yorktown Heights, NY 10598
- 1 DR. PERRY THORNDYKE
THE RAND CORPORATION
1700 MAIN STREET
SANTA MONICA, CA 90406
- 1 Dr. Douglas Towne
Univ. of So. California
Behavioral Technology Labs
1845 S. Elena Ave.
Redondo Beach, CA 90277

Non Govt

- 1 Dr. J. Uhlaner
 Perceptronics, Inc.
 6271 Variel Avenue
 Woodland Hills, CA 91364
- 1 Dr. Benton J. Underwood
 Dept. of Psychology
 Northwestern University
 Evanston, IL 60201
- 1 Dr. Phyllis Weaver
 Graduate School of Education
 Harvard University
 200 Larsen Hall, Appian Way
 Cambridge, MA 02138
- 1 Dr. David J. Weiss
 N660 Elliott Hall
 University of Minnesota
 75 E. River Road
 Minneapolis, MN 55455
- 1 DR. GERSHON WELTMAN
 PERCEPTRONICS INC.
 6271 VARIEL AVE.
 WOODLAND HILLS, CA 91367
- 1 Dr. Keith T. Wescourt
 Information Sciences Dept.
 The Rand Corporation
 1700 Main St.

